# TANDBERG Border Controller
## User Manual




Software version Q3.0
D13691.03

# Trademarks and copyright

# Environmental Issues

Thank you for buying a product which contributes to a reduction in pollution, and thereby helps save the environment. Our products reduce the need for travel and transport and thereby reduce pollution. Our products have either none or few consumable parts (chemicals, toner, gas, paper). Our products are low energy consuming products.

## TANDBERG's Environmental Policy

- TANDBERG's Research and Development is continuously improving TANDBERG's products towards less use of environmentally hazardous components and substances as well as to make the products easier to recycle.
- TANDBERG's products are Communication Solutions. The idea of these solutions is to reduce the need for expensive, time demanding and polluting transport of people. Through people's use of TANDBERG's products, the environment will benefit from less use of polluting transport.
- TANDBERG's wide use of the concepts of outsourcing makes the company itself a company with a low rate of emissions and effects on the environment.
- TANDBERG's policy is to make sure our partners produce our products with minimal influence on the environment and to demand and audit their compatibility according to applicable agreements and laws (national and international).

## Environmental Considerations

Like other electronic equipment, the TANDBERG Border Controller contains components that may have a detrimental effect on the environment. TANDBERG works continuously towards eliminating these substances in our products.

- Printed-wiring boards made of plastic, with flame-retardants like Chloride or Bromide.
- Component soldering that contains lead.
- Smaller components containing substances with possible environmental effect.

After the product's end of life cycle, it should be returned to authorized waste handling and should be treated according to National and International Regulations for waste of electronic equipment.

# Operator Safety Summary

For your protection, please read these safety instructions completely before operating the equipment and keep this manual for future reference. The information in this summary is intended for operators. Carefully observe all warnings, precautions and instructions both on the apparatus and in the operating instructions.

## Warnings

- Water and moisture - Do not operate the equipment under or near water - for example near a bathtub, kitchen sink, or laundry tub, in a wet basement, or near a swimming pool or in areas with high humidity.
- Cleaning - Unplug the apparatus from the wall outlet before cleaning or polishing. Do not use liquid cleaners or aerosol cleaners. Use a lint-free cloth lightly moistened with water for cleaning the exterior of the apparatus.
- Ventilation - Do not block any of the ventilation openings of the apparatus. Install in accordance with the installation instructions. Never cover the slots and openings with a cloth or other material. Never install the apparatus near heat sources such as radiators, heat registers, stoves, or other apparatus (including amplifiers) that produce heat.
- Grounding or Polarization - Do not defeat the safety purpose of the polarized or grounding-type plug. A polarized plug has two blades with one wider than the other. A grounding type plug has two blades and a third grounding prong. The wide blade or third prong is provided for your safety. If the provided plug does not fit into your outlet, consult an electrician.
- Power-Cord Protection - Route the power cord so as to avoid it being walked on or pinched by items placed upon or against it, paying particular attention to the plugs, receptacles, and the point where the cord exits from the apparatus.
- Attachments - Only use attachments as recommended by the manufacturer.
- Accessories - Use only with a cart, stand, tripod, bracket, or table specified by the manufacturer, or sold with the apparatus. When a cart is used, use caution when moving the cart/apparatus combination to avoid injury from tip-over.
- Lightning - Unplug this apparatus during lightning storms or when unused for long periods of time.
- Servicing - Do not attempt to service the apparatus yourself as opening or removing covers may expose you to dangerous voltages or other hazards, and will void the warranty. Refer all servicing to qualified service personnel.
- Damaged Equipment - Unplug the apparatus from the outlet and refer servicing to qualified personnel under the following conditions:
  - When the power cord or plug is damaged or frayed
  - If liquid has been spilled or objects have fallen into the apparatus
  - If the apparatus has been exposed to rain or moisture

- If the apparatus has been subjected to excessive shock by being dropped, or the cabinet has been damaged
- If the apparatus fails to operate in accordance with the operating instructions.

# Contents

# 1 Introduction

This User Manual is provided to help you make the best use of your TANDBERG Border Controller.

A Border Controller is a key component of TANDBERG's Expressway$^{TM}$firewall traversal solution. Used in conjunction with a TANDBERG Gatekeeper or TANDBERG traversal enabled endpoints it allows calls to be made into and out of a secured private network.

The main features of the TANDBERG Border Controller are:

- IPv4 and IPv6 support
- Registration of traversal enabled endpoints.
- Supports up to 500 registered TANDBERG traversal endpoints.
- Secure firewall traversal of any firewall or NAT.
- Up to 100 traversal calls.
- Supports up to 100 neighboring zones.
- Flexible zone configuration with prefix and suffix support.
- URI dialing with DNS enabling global connectivity.
- Can function as a standalone Border Controller or be neighbored with other Border Controllers and Gatekeepers.
- Can be used to control the amount of bandwidth used both within the Border Controller zone and to neighboring Border Controllers and Gatekeepers.
- Can limit total bandwidth usage and set maximum per call bandwidth usage with automatic down-speeding if call exceeds per-call maximum.
- Can be managed with TANDBERG Management Suite 11.0 or newer, or as a standalone system with RS-232, Telnet, SSH, HTTP and HTTPS.
- Embedded setup wizard on serial port for initial configuration.

Note that features may vary depending on software package.

## 1.1 TANDBERG Border Controller Overview

On the front of the Border Controller there are three LAN interfaces, a serial port (Data 1) and an LED showing the power status of the system. The LAN 1 interface is used for connecting the system to your network, LAN interface 2 and 3 are disabled. The serial port (Data 1) is for connection to a PC, and power on is indicated by the Light Emitting Diode (Power) being lit.




The back of the Border Controller has a power connector, a power switch, and a serial port (Data 2) for connecting to a PC.



# 2 Installation

## 2.1 Precautions

- Never install communication equipment during a lightning storm.
- Never install jacks for communication cables in wet locations unless the jack is specifically designed for wet locations.
- Never touch uninstalled communication wires or terminals unless the communication line has been disconnected at the network interface.
- Use caution when installing or modifying communication lines.
- Avoid using communication equipment (other than a cordless type) during an electrical storm.
- There may be a remote risk of electrical shock from lightning.
- Do not use communication equipment to report a gas leak in the vicinity of the leak.
- The socket outlet shall be installed near to the equipment and shall be easily accessible.
- Never install cables without first switching the power OFF.
- This product complies with directives: LVD 73/23/EC and EMC 89/366/EEC.
- Power must be switched off before power supplies can be removed from or installed into the unit.

## 2.2 Unpacking

The TANDBERG Border Controller is delivered in a special shipping box which should contain the following components:

- Border Controller unit
- Installation sheet
- User manual and other documentation on CD
- Rack-ears and screws
- Kit with 4 rubber feet
- Cables:
  - Power cables
  - One Ethernet cable
  - One null-modem RS-232 cable

### 2.2.1 Installation site preparations

- Make sure that the Border Controller is accessible and that all cables can be easily connected.
- For ventilation: Leave a space of at least 10cm (4 inches) behind the Border Controller's rear and 5cm (2 inches) on the sides.
- The room in which you install the Border Controller should have an ambient temperature between 0°C and 35°C (32°F and 95°F) and between 10% and 90% non-condensing relative humidity.
- Do not place heavy objects directly on top of the Border Controller.
- Do not place hot objects directly on top, or directly beneath the Border Controller.
- Use a grounded AC power outlet for the Border Controller.

## 2.3 Mounting

The Border Controller comes with brackets for mounting in standard 19" racks.

Before starting the rack mounting, please make sure the TANDBERG Border Controller is placed securely on a hard, flat surface.

1. Disconnect the AC power cable.
2. Make sure that the mounting space is according to the 'Installation site preparations' in section 2.2.1.
3. Attach the brackets to the chassis on both sides of the unit.
4. Insert the unit into a 19" rack, and secure it with screws.

## 2.4 Connecting Cables

**Power cable** Connect the system power cable to an electrical distribution socket.

**LAN cable** Connect a LAN cable from the LAN 1 connector on the front of the unit to your network.

**Null-modem RS-232 cable** Connect the supplied null-modem RS-232 cable between the Border Controller's Data 1 connector and the COM port on a PC.

## 2.5 Switching on the System

To start the TANDBERG Border Controller, make sure that the following has been done:

- The power cable is connected.
- The LAN cable is connected.

Then switch the power switch button on the back of the unit to ‘1’.

On the front of the chassis you will see the Power LED being lit.

## 2.6 Border Controller Initial Configuration

The TANDBERG Border Controller requires some configuration before it can be used. This must be done using a PC connected to the serial port (Data 1) or by connecting to the system’s default IP address: 192.168.0.100.

The IP address, subnet mask and gateway must be configured before use. The Border Controller has to be configured with a static IP address. Consult your network administrator for information on which addresses to use.

To set the initial configuration, do the following:

1. Connect the supplied null-modem RS-232 cable from Data 1 to a PC running a terminal program.
2. Start a terminal program and configure it to use the serial port with baud rate 115200, 8 data bits, no parity, 1 stop bit, no flow control.
3. Power on the unit if it is not already on.
4. You should see the unit display start up information.
5. After approximately 2 minutes you will get a login prompt.
6. Enter username *admin* and your password. The default password is *TANDBERG*.
7. You will be prompted if you want to run the install wizard. Type *y* and press Enter.

```
(none) login:  admin
Password:
Run install wizard [n]:  y
```

8. Specify the following:
   (a) The password you want to use for your system. See section 3.1.1 for account details.
   (b) The IP address of the system.
   (c) The IP subnet mask of the system.
   (d) The IP default gateway of the system.
   (e) The Ethernet speed.
   (f) The local zone prefix, if any, you want to use for the zone controlled by this system.
   (g) Whether you want to use SSH to administer the system.
   (h) Whether you want to use Telnet to administer the system.
9. You will be prompted to login again. You should see a welcome message like this:

```
Welcome to
TANDBERG Border Controller Release Q3.0
SW Release Date:  2006-01-02
OK
```

10. Login with username *admin* and your password.
11. Review other system settings. You may want to set the following:
    (a) The name of the Border Controller. This is used to identify the Border Controller by the TANDBERG Management Suite. See the `xConfiguration SystemUnit` command in section 11.2.17 for more information on setting the name.
    (b) Automatic discovery. If you have multiple Border Controllers in the same network you may want to disable automatic discovery on some of them. See the xConfiguration Gatekeeper AutoDiscovery command in section 11.2.4.
    (c) The DNS server address and the domain name if the Border Controller will be configured with hostnames instead of IP address or if URI dialing is required. See the xConfiguration DNS Server Address command in section 11.2.6 for more information.
12. Reboot the Border Controller by typing the command `xCommand boot` to make your new settings take effect.
13. Disconnect the serial cable.

**NOTE** To securely manage the Border Controller you should disable HTTP and Telnet, using the encrypted HTTPS and SSH protocols instead. For increased security, disable HTTPS and SSH as well, using the serial port to manage the system.

**NOTE** If you do not have an IP gateway, configure the Border Controller with an unused IP address that is valid in your subnet.

# 3 Getting started

## 3.1 System Administration

To configure and monitor the TANDBERG Border Controller you can either use the web interface or a command line interface. The command line interface is available over SSH and Telnet, or through the serial port. The interface is the same using all three access methods. By default administration sessions remain active until you logout. Session timeouts may be enabled using the `xConfiguration Session TimeOut` command.

To enter commands you should start a session and login with user name *admin* and your password.

The interface groups information in different commands:

**xstatus** Provides a read only interface to determine the current status of the system. Information such as current calls and registrations is available through this command group.

**xconfiguration** A read/write interface to set system configuration data such as IP address and subnet.

**xcommand** A miscellaneous group of commands for setting information or obtaining it.

**xhistory** Provides historical information about calls and registrations.

**xfeedback** An event interface, providing information about calls and registrations.

A command reference is given in section 11

### 3.1.1 Administrator Account

All administration requires you to log in to the administration account with a user name `admin` and a password. The default password is TANDBERG, which you are recommended to change as soon as possible. Choose a strong password, particularly if administraion over IP is enabled. Changing the password can only be done through the command line interface using the command:

`xconfiguration systemunit password:` *new_password*

If you forget your password, it is possible to set a new password using the following procedure:

- Reboot the Border Controller.
- Connect to the Border Controller over the serial interface once it has restarted.
- Login with the user name `pwrec`. No password is required.
- You will be prompted for a new password.

The `pwrec` account is only active for one minute following a restart. Beyond that time you will have to restart the system again to change the password. Because access to the serial port allows the password to be reset, it is recommended that you install the Border Controller in a physically secure environment.

### 3.1.2 Root Account

The Border Controller provides a root account with the same password as the admin account. This account should not be used in normal operation, and in particular system configuration should not be conducted using this account: use the admin account instead.

## 3.2 Registration

Before an endpoint can use the Border Controller it must first register with it. There are two ways an endpoint can register:

- Automatically.
- Manually by specifying the IP address of the Border Controller.

You can disable automatic registration on the Border Controller. See auto discovery in section 11.2 for more information.

When registering, the endpoint registers with one or more of the following:

- One or more H.323 IDs.
- One or more E.164 aliases.

Users of other registered endpoints can then call the endpoint by using either the H.323 ID, a URI, an E.164 alias, or one of the services.

You should choose H.323 aliases which do not reveal sensitive information. Like e-mail addresses, they are passed unencrypted when a call is made.

Consult the endpoint documentation for information on how to configure it with a Gatekeeper.

**NOTE** Only traversal enabled endpoints can register with a TANDBERG Border Controller. All other registration requests will be rejected. Traversal enabled endpoints include all TANDBERG Expressway endpoints and third party endpoints which support the ITU H.460.18 and H.460.19 standards.

**NOTE** When URI dialing is used to discover an endpoint, the URI used is based on either the H.323 ID or the E.164 alias that the endpoint registered with. The local domain is then added to this. See section 6

## 3.3 Neighbor Gatekeepers

As you start deploying more than one Gatekeeper or Border Controller, it is useful to neighbor the systems together so that they can exchange information about registered endpoints. Each Gatekeeper or Border Controller forms an H.323 zone and is responsible for the endpoints within that zone.

The simplest approach is to assign each endpoint a unique alias and divide the endpoint registrations between the Gatekeepers and Border Controllers. Each Gatekeeper or Border Controller is then configured with the addresses of all other Gatekeepers and Border Controllers. When a system receives a call for an endpoint which is not registered with, it will send out a

Location Request to all the other Gatekeepers and Border Controllers on the system. Whilst conceptually simple, this sort of flat dial plan does not scale very well: adding or moving a Gatekeeper requires changing the configuration of every Gatekeeper and Border Controller; one call attempt can result in a flood of location requests. With 3 interconnected systems, an unknown alias can result in 28 LRQs, with 4 systems some 50,000 LRQs will be generated.

An alternative deployment would use a structured dial plan: endpoints are assigned an alias based on the system they are registering with. Using E.164 aliases, each Gatekeeper or Border Controller would be assigned an area code. When the Gatekeepers and Border Controllers are neighbored together, each neighbor is configured with its corresponding area code as a prefix. That neighbor will now will only be queried for calls to numbers which begin with its prefix. In a URI based dial plan, similar behaviour may be obtained by configuring neighbors with a suffix to match the desired domain name.

It may be desirable to have endpoints register with just the subscriber number — the last part of the E.164 number. In that case, the Border Controller should be configured to strip prefixes before placing the Location Request.

A structured dial plan will minimize the number of location requests issued when a call is attempted, but, as described above, still requires a fully connected mesh of all Gatekeepers and Border Controllers in your deployment. A hierarchical dial plan can simplify this. One Gatekeeper is nominated as the directory gatekeeper for the deployment. All Border Controllers and public Gatekeepers are neighbored with it and vice versa. There is no need to neighbor the Border Controllers and public Gatekeepers with each other. Adding a new Border Controller or public Gatekeeper now only requires changing configuration on that system and the Directory Gatekeeper.

Failure of the directory gatekeeper could cause significant disruption to communications. Consideration should be given to the use of Alternate Gatekeepers (section 3.4) for increased resilience.

Neighbors are added and zones configured through the command line interface using the `xconfiguration zones` family of command, `xCommand ZoneAdd` or through the web interface: *Border Controller Configuration* → *Zones* as shown in Figure 1. The prefixes and suffixes described above are formed using patterns: each zone may have up to 5 patterns assigned, each of which may be defined as a prefix or a suffix.

### 3.3.1 Search Order

If a called alias matches a prefix or suffix zone a strong match is achieved. A weak match is achieved if a zone is to be queried only because it has no pattern matching configured.

When an incoming call request is received a Border Controller will first search all of its registered endpoints. If no match is found, all strongly matching neighbor and traversal zones will be queried concurrently. If the target is not found in any of the strongly matching zones, all weakly matching neighbor zones will be queried, then all weakly matching traversal zones. Finally, if a match has still not been found, a DNS query may be attempted as described in section 6.

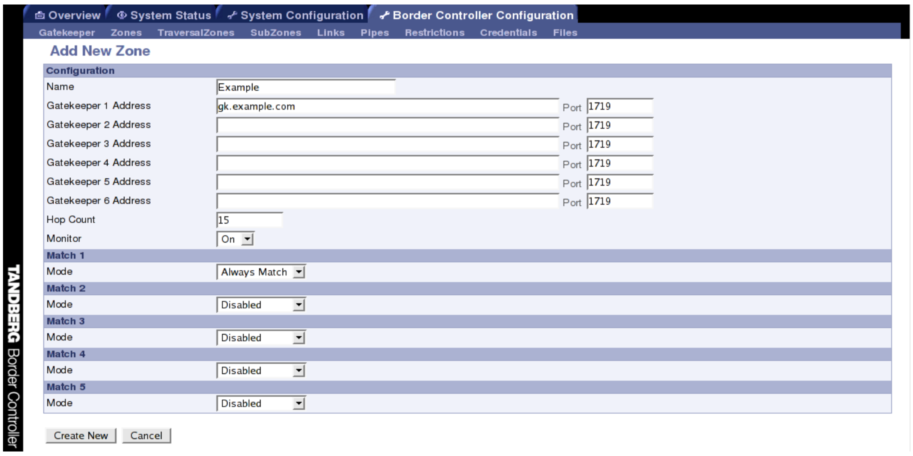



Figure 1: Adding a new zone

## 3.4 Alternate Border Controllers

Alternate Border Controller support is provided to increase the reliability of your deployment. If one Border Controller becomes unavailable, perhaps due to a network or power outage, another will be used as an Alternate. Alternates share responsibility for their endpoint community: an individual endpoint may be registered with any one of the Alternates. You should configure Alternates identically for all registration and call features such as authentication, bandwidth control and policy. If you do not do this, endpoint behavior will vary unpredictably depending on which Alternate it is currently registered with. Alternates should also be deployed on the same LAN as each other so that they may be configured with the same routing information such as local domain names and local domain subnet masks.

Each Border Controller may be configured with the IP addresses of up to five Alternates. When an endpoint registers with the Border Controller, it is presented with the IP addresses of all the Alternates. If the endpoint loses contact with its initial Border Controller, it will seek to register with one of the Alternates. This may result in your endpoint community's registrations being spread over all the Alternates.

Enterprise Gatekeepers which register with the Border Controller may also be given a list of Alternate Border Controllers to use.

When a Border Controller receives a Location Request, if it cannot respond from its own registration database, it will query all of its Alternates before responding. This allows the pool of registrations to be treated as if they were registered with a single Border Controller.

The Alternate Border Controllers can be configured within the web interface of the Border Controller by navigating to *Border Controller Configuration → Gatekeeper*. Up to five different alternates can be configured. Please see Figure 2 for a screenshot of a sample configuration.

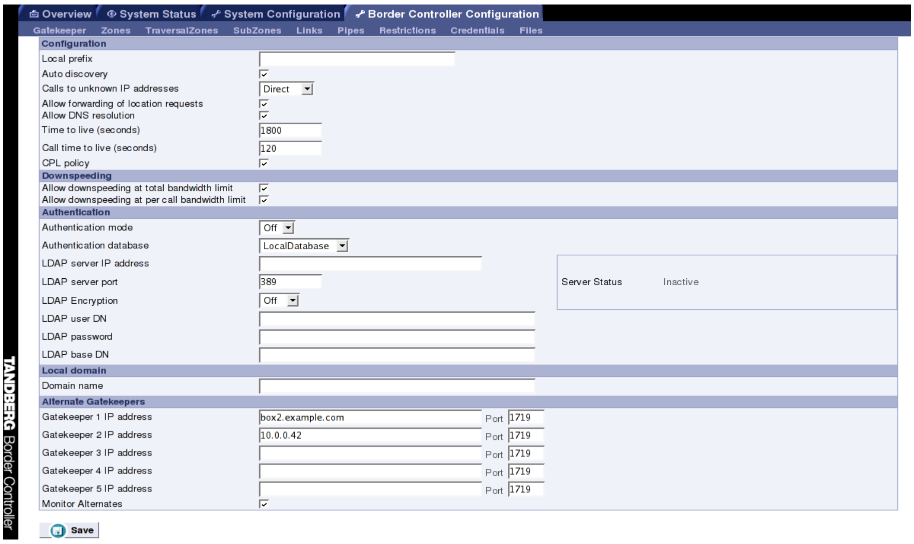


Figure 2: Alternate Border Controller configuration

## 3.5 Call Control

When an endpoint wants to call another endpoint it presents the address it wants to call to the Border Controller using a protocol knows as RAS. The Border Controller tries to resolve this address and supplies the calling endpoint with information about the called endpoint. The destination address can take several forms: IP address, H.323 ID, E.164 alias or a full H.323 URI.

When an H.323 ID or E.164 alias is used, the Border Controller looks for a match between the dialed address and the aliases registered by its endpoints. If no match is found, it may query other Gatekeepers and Border Controllers.

When dialing by H.323 URI, the destination address resembles an email address. The Border Controller first follows the procedure for matching H.323 IDs. If that fails it looks for a Gatekeeper or Border Controller responsible for the domain (the part of the URI following the @ symbol) and queries that device.

Dialing by IP address is necessary when the destination endpoint is not registered with a Gatekeeper or Border Controller. If it is registered, then one of the other addressing schemes should be used instead as they are more flexible. From your registered endpoint, dial the IP address of the endpoint you wish to call. This requires that the Border Controller has `xConfiguration Gatekeeper CallToUnknownIPAddresses` correctly configured.

It is not possible to dial endpoints behind a Border Controller by IP address. Calls should be made using an E.164 or H.323 alias.

Figure 3 illustrates the process the Border Controller performs when receiving call requests:

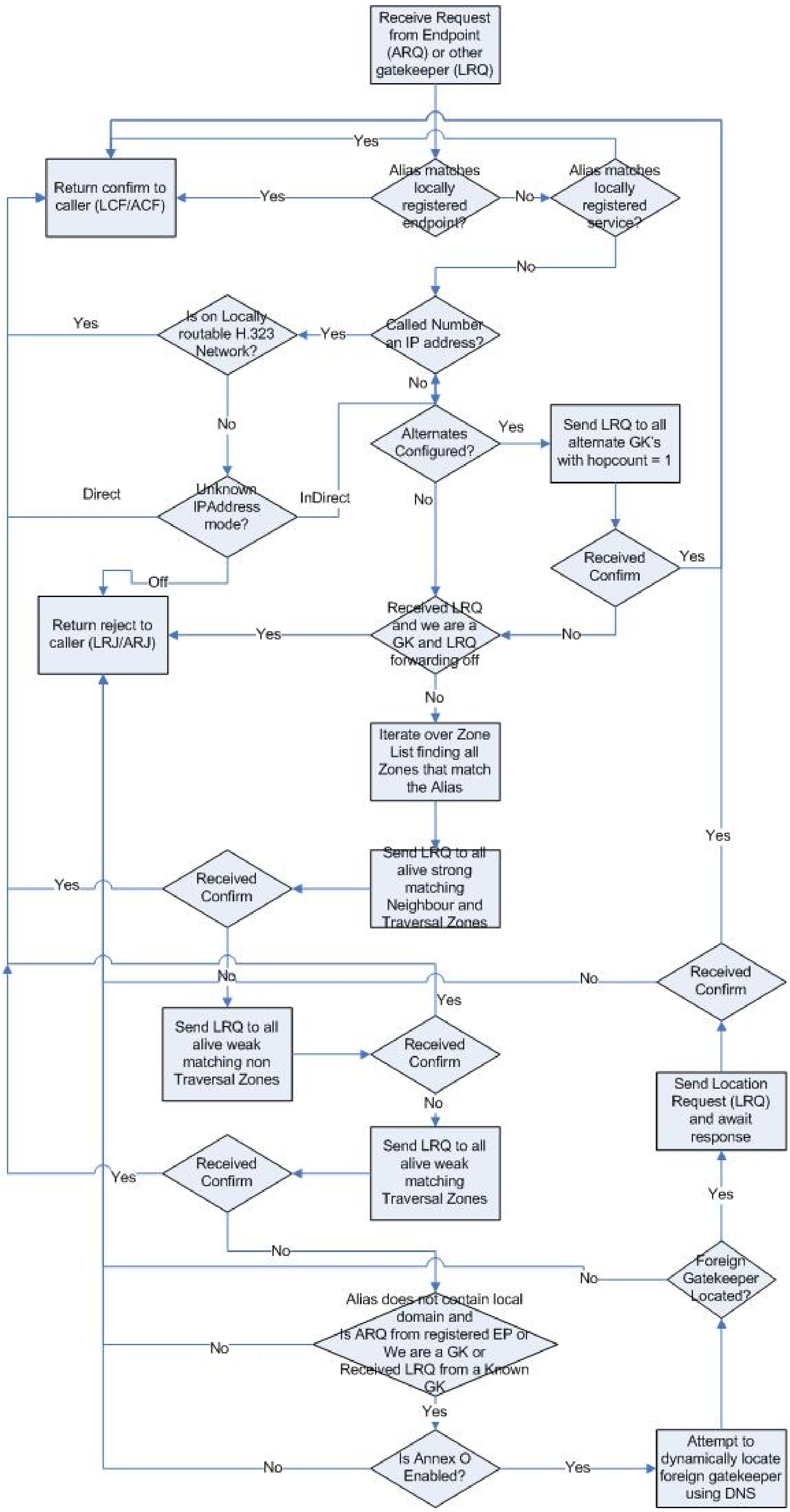



Figure 3: Location decision flow diagram

## 3.6 Firewall Traversal

The Border Controller works with the TANDBERG Gatekeeper, TANDBERG Expressway endpoints and other endpoints which support the ITU H.460.18 and H.460.19 standards. In order to successfully traverse a firewall, the firewall is required to allow initial outbound traffic to designated ports on the border controller and return traffic from those ports. The ports used are configurablea and by default are:

- UDP/1719
- TCP/1720
- TCP/2776
- TCP/2777
- UDP/2776
- UDP/2777

Non traversal calls — calls to the public internet — send traffic to ports determined by the receiving endpoint and from ports. Traffic is sent from UDP ports 1719 and 50,000–51,000 and TCP ports 15,000–24,000

Having the firewall only accept incoming data from the IP address and port to which data has already been sent allows you to maintain a secure network behind the firewall: unsolicited incoming data will not be accepted.

You are recommended to turn off any H.323 traversal features on the firewall: these are not needed in conjunction with the Expressway solution and may interfere with its operation.

The Gatekeeper identifies itself to the Border Controller with its Traversal Zone Name which may be determined with the command:

```
xConfiguration Zones TraversalZone Name
```

or using the Gatekeeper's web interface on the *System Configuration* → *Misc* page.

Up to 50 Gatekeepers may register with the Border Controller. Each is identified with a unique Traversal Zone Name which is set with the command:

```
xConfiguration Zones TraversalZone [1..50 ] Name: name
```

or using the Border Controller's web interface on the *Border Controller Configuration* → *TraversalZones* page.

# 4 Bandwidth Control

The TANDBERG Border Controller allows you to control endpoints' use of bandwidth on your network. Figure 4 shows a typical deployment: a broadband LAN, where high bandwidth calls are acceptable, a pipe to the internet with restricted bandwidth, and two satellite offices, each with their own restricted pipes. In order to utilize the available bandwidth efficiently, the TANDBERG Border Controller allows you to model your network, and bandwidth controls on individual components of the network. Bandwidth controls may be set on a call by call basis and on a total concurrent usage basis.

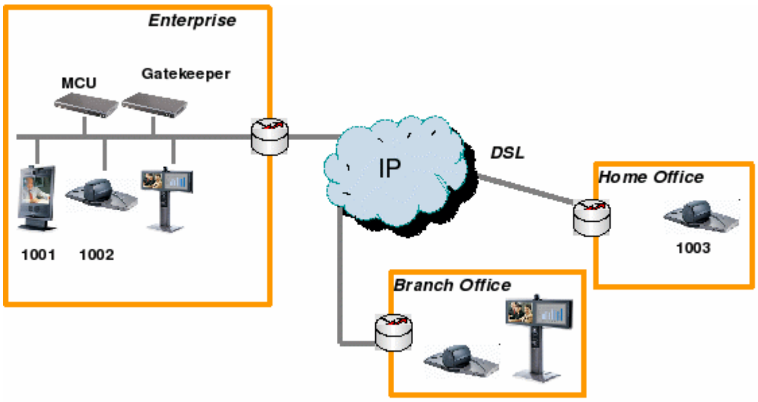



Figure 4: Typical network deployment

All endpoints registered with your Border Controller are part of its local zone. As shown in Figure 4, the local zone can contain many different networks with different bandwidth limitations. In order to model this, the local zone is made up of one or more subzones. When an endpoint registers with the Border Controller it is assigned to a subzone, based on its IP address.

By default all endpoints registering with the Border Controller are assigned to the default subzone. This is suitable if you have uniform bandwidth available between all your endpoints. When you have differing bandwidth provision, as in Figure 4, you should create a new subzone for each pool of endpoints.

Subzones are added and configured through the web interface on the *Border Controller Configuration* → *SubZones* page (Figure 5), or through the command line using the following commands:

```
xConfiguration SubZones SubZone [1..100] Name
xConfiguration SubZones SubZone [1..100] Subnet IP Prefixlength
xConfiguration SubZones SubZone [1..100] Subnet IP Address
```

Subzones may be configured with links joining them to each other and to other zones. These links are used to calculate how a call is routed over the network and so which zones and subzones are involved. If multiple routes are possible, your Border Controller will select the one with the fewest links.

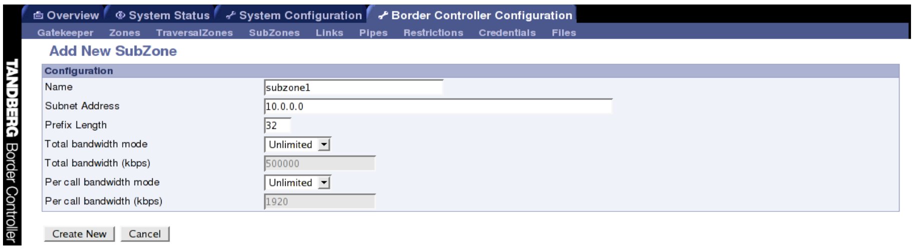



Figure 5: Configuring a SubZone

Links may be configured through the web interface on the *Border Controller Configuration* → *Links* page, or through the command line using the following commands:

```
xConfiguration Links Link [1..100] Name
xConfiguration Links Link [1..100] Node1 Name
xConfiguration Links Link [1..100] Node2 Name
xConfiguration Links Link [1..100] Pipe1 Name
xConfiguration Links Link [1..100] Pipe2 Name
```

Each subzone may be configured with its own bandwidth limits. Calls placed between two endpoints in the same subzone consume resource from the subzone's allocation. Subzone bandwidths are configured on the *Border Controller Configuration* → *SubZones* page (see Figure 6 for a screenshot of the configuration) or using the following command line commands:

```
xConfiguration SubZones SubZone [1..100] Bandwidth Total Mode
xConfiguration SubZones SubZone [1..100] Bandwidth Total Limit
xConfiguration SubZones SubZone [1..100] Bandwidth PerCall Mode
xConfiguration SubZones SubZone [1..100] Bandwidth PerCall Limit
```

When calls are placed between endpoints in different subzones, it is possible to control the bandwidth used on the link between them. To do this, create a pipe and configure it with the required bandwidth characteristics. This pipe is then assigned to a link. Calls traversing the link will now take the pipe's bandwidth allocation into consideration. Pipes are created and configured on the *Border Controller Configuration* → *Pipes* page (Figure 6) or using the following command line commands:

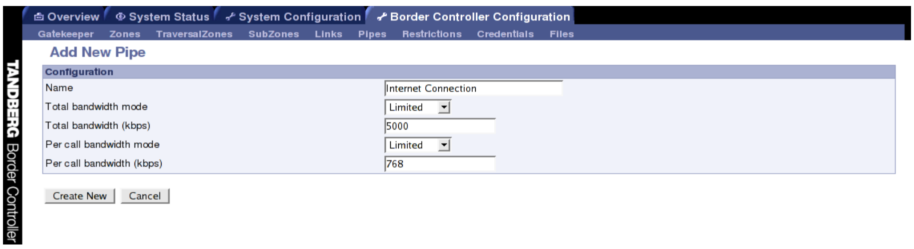



Figure 6: Configuring a pipe

```
xConfiguration Pipes Pipe [1..100] Name
xConfiguration Pipes Pipe [1..100] Bandwidth Total Mode
xConfiguration Pipes Pipe [1..100] Bandwidth Total Limit
```

```
xConfiguration Pipes Pipe [1..100] Bandwidth PerCall Mode
xConfiguration Pipes Pipe [1..100] Bandwidth PerCall Limit
```

Pipes may be shared between one or more links. This is used to model the situation where a site communicates with several other sites over the same broadband connection to the Internet. Each link may have up to two pipes associated with it. This is useful for modeling two sites, each with their own broadband connection to the Internet backbone. Calls between zones or subzones consume bandwidth from each zone and any pipes on the link between them.

When a Border Controller is neighbored with another Gatekeeper or a Border Controller, the neighbor is placed in its own zone. This allows you to control the bandwidth used by calls to and from endpoints controlled by the other Gatekeeper. Sometimes you may place and receive calls to Gatekeepers you are not neighbored with (See section 6). These Gatekeepers, and any unregistered endpoints reached by dialing their IP address, are placed in the Default Zone.

If bandwidth control is in use, there are two possible behaviors when a call cannot be placed at the bandwidth requested. By default the call will be connected at a reduced bandwidth (down-speeding), assuming that there is some bandwidth still available. Optionally the call may be rejected if it cannot be placed at the requested bandwidth. This option is controlled through the web interface of the Border Controller by navigating to *Border Controller Configuration* → *Gatekeeper*(Figure 7) or through the following command line instructions:




Figure 7: Configuring down-speeding options

```
xConfiguration Gatekeeper Downspeed PerCall Mode:  <On/Off>
xConfiguration Gatekeeper Downspeed Total Mode:  <On/Off>
```

## 4.1 Bandwidth Control and Firewall Traversal

When a Border Controller and Gatekeeper are being used to traverse a firewall, an additional zone and subzone come into use.

The traversal zone is used to represent the zone containing the Gatekeeper Controller this Border Controller is paired with. This zone is automatically added for you. The traversal subzone represents the Border Controller itself. The traversal subzone allows you to control total and per call bandwidths passing through the Border Controller. Unlike other subzones, no endpoints will ever be registered in this subzone.

## 4.2 Bandwidth Control Examples

One possible configuration for the deployment in Figure 4 is shown in Figure 8. Each of the offices is represented as a separate subzone, with bandwidth configured according to local policy. The enterprise's leased line connection to the Internet, and the DSL connections to the remote offices, are modelled as separate pipes.

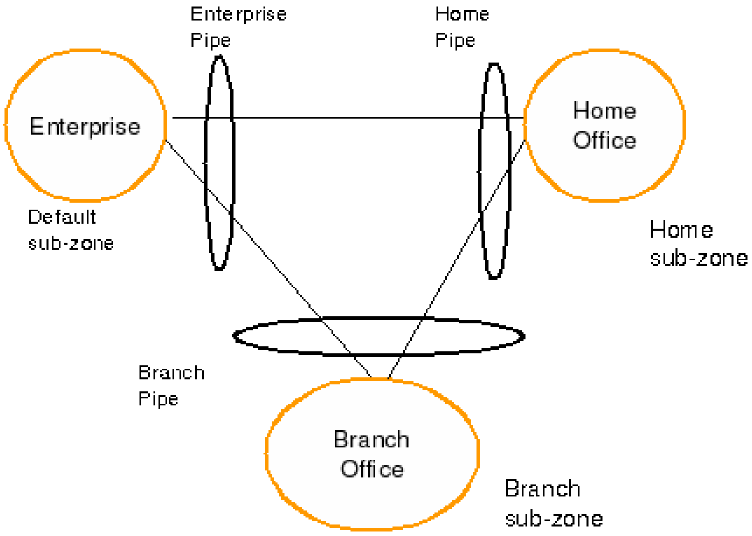



Figure 8: Bandwidth control example

There are no firewalls involved in the scenario shown in figure 4, so we can configure links between each of the offices. Each link is then assigned two pipes, representing the Internet connections of the offices at each end of the link. A call placed between the Home Office and Branch Office will consume bandwidth in the home and branch subzones and on the home and branch pipe. The enterprise's bandwidth budget will be unaffected by the call.

If we now modify our deployment to include firewalls between the offices, we can use the firewall traversal capability of the TANDBERG Gatekeeper and Border Controller to maintain connectivity.

In Figure 9, the endpoints in the enterprise register with the Gatekeeper, whilst those in the branch and home office register with the Border Controller.

Figure 10 shows how the Border Controller could be configured for the deployment in Figure 9. The introduction of the firewalls means that there is no longer any direct connectivity between the Branch and Home offices. All traffic must be routed through the Border Controller. This is shown by the absence of a link between the Home and Branch subzones.

The Traversal Zone in Figure 10 represents the Enterprise Gatekeeper. The Border Controller will consume bandwidth from the Traversal Zone for all calls placed to endpoints managed by the Enterprise Gatekeeper. In this example we have assumed that there is no bottleneck on the link between the Border Controller and the Enterprise network, so have not placed a pipe on this link. If you want to limit the amount of traffic flowing through your firewall, you could provision a pipe on this link.

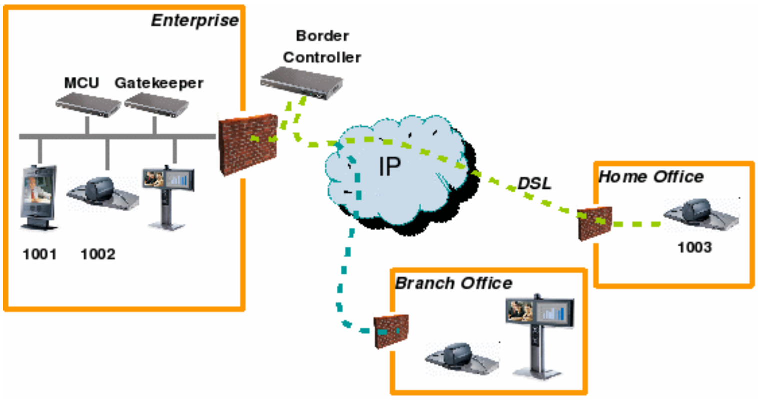



Figure 9: Network Deployment with firewalls

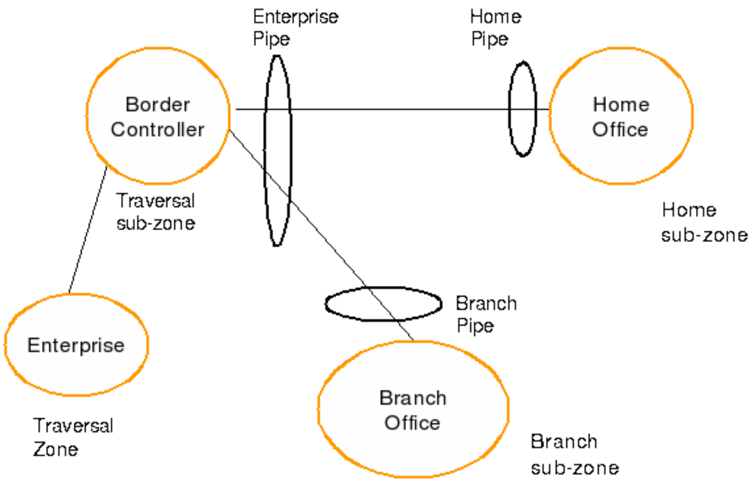



Figure 10: Border Controller example configuration

The traversal subzone in Figure 10 may be used to control the amount of traffic flowing through the Border Controller itself.

Because the Gatekeeper is only managing endpoints on the LAN, its configuration is simpler as shown in Figure 11.

All of the endpoints in the enterprise will be assigned to the default subzone. The Traversal subzone controls traversal traffic flowing through the Gatekeeper, whilst the Traversal Zone controls all traffic traversing the enterprise firewall and passing on to the Border Controller. Both subzones and the Traversal zone are linked: the link between the default subzone and the Traversal zone is used by endpoints which can send media directly to the Border Controller. The other two links are used by endpoints using the Gatekeeper to traverse the firewall.

The Border Controller is shipped with Default Zone and Default and Traversal subzones already configured. They are also preconfigured with the links between these zones to allow calls to be placed. You may delete or amend the default links if you need to model restrictions of your

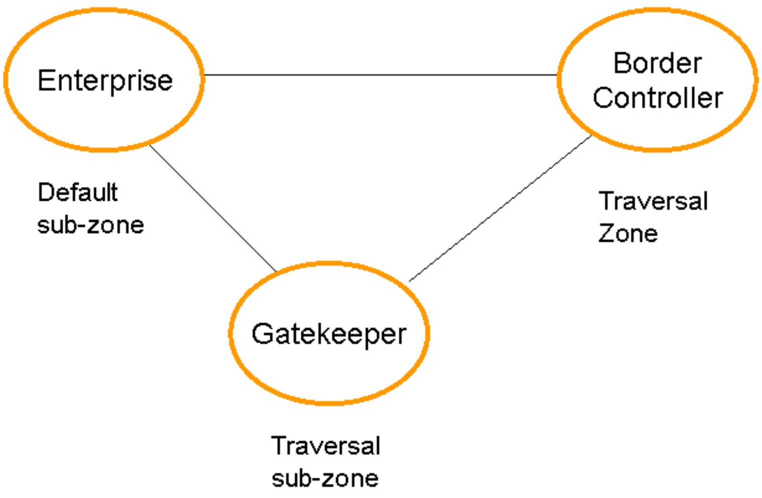



Figure 11: Gatekeeper example configuration

network. The default links may be restored by running the command:

```
xCommand DefaultLinksAdd
```

# 5 Registration Control

The TANDBERG Border Controller can control which endpoints are allowed to register with it. Two separate mechanisms are provided: a simple Registration Restriction Policy and an authentication process based on user names and passwords. It is possible to use both mechanisms at once: authentication to verify an endpoint's identity from a corporate directory and registration restriction to control which of those authenticated endpoints may register with a particular Border Controller.

## 5.1 Registration Restriction Policy

When an endpoint registers with your Border Controller it presents a list of aliases. By default, registration restriction policy is set to None. In this state, any endpoint may register. The registration restriction policy can be configured using the following command:

```
xConfiguration Gatekeeper Registration RestrictionPolicy [None|AllowList|DenyList
]
```

or by using the web interface, on the *Border Controller Configuration* → *Restrictions page* (see Figure 12 for a screenshot of the Registration Restrictions Configuration). If the policy is set to AllowList, only those endpoints with an alias which matches an entry in the AllowList may register. Conversely, if the policy is set to DenyList, all endpoints may register, unless they match an entry on the DenyList. Allow lists and Deny lists are mutually exclusive: only one may be in use at any given time.




Figure 12: Configuring registration restrictions

Matching uses a simple form of wild card expansion:

| | |
|---|---|
| 12345678 | Exact match only |
| 1234567? | First 7 characters are an exact match, last may be anything |
| 123* | 123 followed by anything |
| *@example.com | Any string ending with @example.com |

To set entries in the Allow and Deny lists use the following commands `AllowListAdd, AllowListDelete, DenyListAdd, DenyListDelete`

To view the entries in the allow and deny lists, use the following commands:

```
xConfiguration Gatekeeper Registration AllowList
xConfiguration Gatekeeper Registration DenyList
```

## 5.2 Authentication

The TANDBERG Border Controller can use a user name and password based challenge-response scheme to permit registrations. For details of how to configure your endpoint with the appropriate information, please consult your endpoint manual.

The Border Controller supports the ITU H.235 [1] specification for authenticating the identity of network devices with which theBorder Controller communicates.

In order to verify the identity of a device, the Border Controller needs access to the password information. This credential information may be stored in a local database on the Border Controller or obtained from an LDAP Director Server.

### 5.2.1 Authentication using a local database

To configure the Border Controller to use the local database of credentials during authentication issue the following commands

```
xConfiguration Authentication Mode:  On
xConfiguration Authentication Database:  LocalDatabase
```

Each credential in the local database has a username and a password. To manage the credentials in the local database use the following commands

```
xcommand CredentialAdd <user name> <password>
xcommand CredentialDelete <credential index>
```

To show the credentials in the local database use the command

```
xConfiguration Authentication Credential
```

The credential database can also be configured via the web interface on the *Border Controller Configuration* → *Credentials* page (Figure 13).

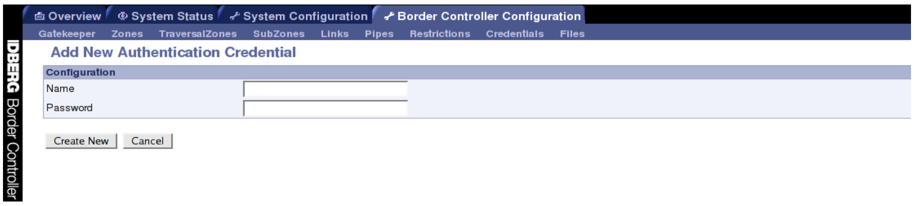



Figure 13: Adding LDAP credentials

### 5.2.2 Authentication using an LDAP server

The authentication information can be obtained from an LDAP server. The directory on the LDAP server should be configured to implement the ITU H.350 specification to store H.235 credentials for devices that the Border Controller communicates with. The directory should also be configured with the H.323 aliases of endpoints that will register with the Border Controller.

For instructions on how to configure common third party LDAP servers, see Appendix B.

To configure the Border Controller to use the LDAP server directory during authentication issue the following commands:

```
xConfiguration Authentication Mode:  On
xConfiguration Authentication Database:  LDAPDatabase
```

The Border Controller needs to be configured with the area of the directory which will be searched for the communication device information. This should be specified as the Distinguished Name (DN) in the directory under which the H.350 objects reside:

```
xConfiguration Authentication LDAP BaseDN: "Your base DN"
```

The Border Controller must also be configured with the location of the LDAP server and the security credentials required to gain access to the LDAP server. The following commands are used to configure the LDAP server details:

```
xConfiguration LDAP Server Address:  "ldap server address"
xConfiguration LDAP Server Port:  389
xConfiguration LDAP UserDN: "Your user DN"
xConfiguration LDAP Password:  "password"
```

The status of the connection between the Border Controller and the LDAP server can be verified using the command:

```
xstatus LDAP
```

The details of the LDAP server can also be configured via the web interface on the *Border Controller Configuration* → *Gatekeeper* page).

### 5.2.3 Securing the LDAP connection with TLS

The traffic between the Border Controller and the LDAP server can be encrypted using Transport Layer Security (TLS). To use TLS, the LDAP server must have a valid certificate installed so that the Border Controller can verify the server's identity. For more information on setting up certificates using common LDAP servers, see Appendix B LDAPS uses port 636 as its default communications port.

Using the terminal interface TLS can be enabled with the following command

```
xConfiguration LDAP Encryption:  TLS
```

TLS can also be enabled via the web interface using the *Border Controller Configuration* → *Gatekeeper* page.

The Border Controller will now only communicate with the LDAP server using TLS. To verify the identity of the LDAP server, the certificate of the Certificate Authority (CA) that issued the LDAP server with its certificate must be uploaded to the Border Controller. To install the CAs certificate, navigate to the *Border Controller Configuration* → *Files* page and upload the CA certificate as a Trusted CA certificate.

# 6 URI Dialing

If an alias is not located in the Border Controller's list of registrations, it may attempt to find an authoritative Gatekeeper through the DNS system.

URI dialing makes it easier for endpoints registered with different Gatekeepers or Border Controllers to call each other. Without URI dialing, you need to neighbor all the systems to each other. This does not scale well as the number of systems grows. It is also inconvenient for making one off calls to endpoints registered with previously unknown systems.

Using URI dialing, you call using an H.323 URI which looks like an email address. The destination Gatekeeper is found from the domain name — the part after the @ — in the same way that an email server is found.

The decision as to whether or not to use URI dialing is governed by the current state of:

```
xConfiguration Gatekeeper DNSResolution Mode:  <On/Off>
```

or using the web interface on the *Border Controller Configuration → Gatekeeper* page

You will also need to configure a DNS server for the systems to query. This is set using:

```
xConfiguration IP DNS Server 1 Address:  <address>
```

or using the web interface on the *System Configuration → IP* page (see Figure 14 for the IP Configuration screen).

If you want others to be able to reach you using URI dialing, add a record to your DNS information as described in Appendix: A

Endpoints will typically register with the Border Controller without their domain name. The Border Controller needs to match a request for *fred@example.com* to a registration for *fred*. To do this, it must be configured with the name of the domain in which its endpoints belong. This is set using

```
xConfiguration Gatekeeper LocalDomain DomainName:  <name>
```

If URI dialing is being used in conjunction with firewall traversal, `DNSResolution Mode` should only be enabled on the Border Controller and on any Gatekeepers on the public network. The DNS records should be updated with the address of the Border Controller as the authoritative Gatekeeper for the enterprise. This ensures that calls placed using URI dialing enter and leave the enterprise through the Border Controller, allowing successful traversal of the firewall.

The LocalDomain DomainName should be set on both the Gatekeeper and the Border Controller. Any Alternates should also have the same LocalDomain Domain Name.

## 6.1 Creating DNS SRV records

URI dialing relies on the presence of SRV Record in the DNS information for the zone. The SRV record specifies the location of a server for a particular protocol and domain. Its format is defined by an Internet standard [3] as

```
Service._Proto.Name TTL Class SRV Priority Weight Port Target
```

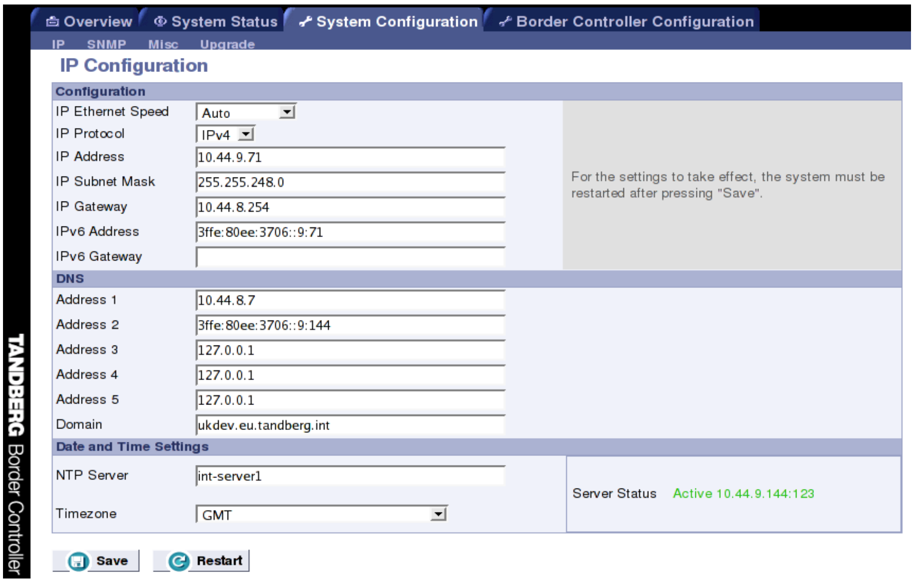



Figure 14: Configuring IP interface

In our case `_Service` is defined by the H.323 protocol suite to be `_h323ls` and `_Proto` is `_udp`. *Name* corresponds to the host part of the H.323 URI.

How you add the SRV record depends on the type of DNS server you are using. Instructions for setting up two common DNS servers are given in Appendix A

# 7 Example Traversal deployments

## 7.1 Simple Enterprise deployment

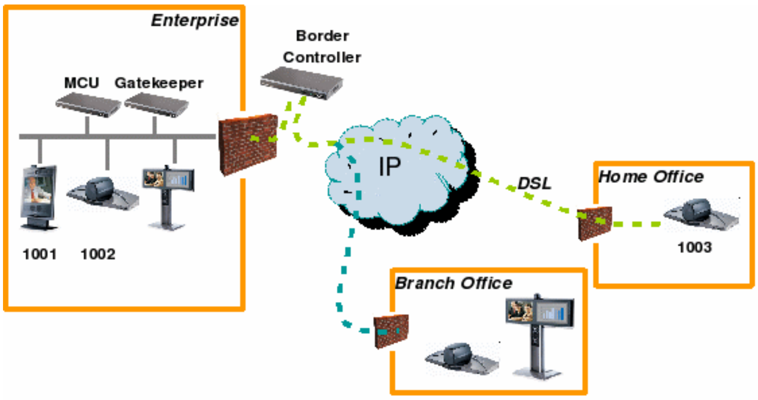



Figure 15: Simple enterprise deployment

Figure 15 shows a typical enterprise deployment. Endpoints 1001, 1002 and a Gatekeeper are deployed on a private network, separated from the public network by a firewall and NAT. Endpoint 1003 is on a separate private network, perhaps a home worker on an DSL connection. A Border Controller is deployed on the public network to allow traversal across the firewalls.

Endpoints 1001, 1002 may be any H.323 compliant endpoint. They will use the TANDBERG Gatekeeper to provide firewall traversal. Endpoint 1003 must be a TANDBERG endpoint which provides firewall traversal.

Endpoints 1001, 1002 should register with the Gatekeeper. Endpoint 1003 will register with the Border Controller. The Gatekeeper will be configured to register with the Border Controller, and the Border Controller set with a traversal client name that matches the TraversalZone Name of the Gatekeeper.

If you wish to be able to call using URI dialing in this deployment then the following configuration is required.

- Enter the address of your DNS server on the Border Controller:

  ```
  xConfiguration DNS Server Address: dns_server_ip_address
  ```

- Enable URI dialing on the Border Controller

  ```
  xConfiguration Gatekeeper DNSResolution Mode: On
  ```

- Ensure that URI dialing is disabled on the Gatekeeper. This is because you wish calls to be routed from the private network to the Border Controller in order to traverse the firewall.

  ```
  xConfiguration URI Dialing Mode: Off
  ```

In order to be able to receive calls placed to *example.com* using URI dialing, configure the following:

- Set *example.com* as the domain name you are using on both the Gatekeeper and Border Controller.
- Update the DNS entry for *example.com* with an A record representing the Border Controller and an SRV record which returns the Border Controller's A record as described in section 6.1

## 7.2 Enterprise Gatekeepers

When an enterprise has already deployed a Gatekeeper to manage calls within the private network, it may be desirable to deploy a traversal solution without having to alter the existing deployment.

In order to achieve this, the TANDBERG Gatekeeper is neighbored with the existing enterprise Gatekeeper as shown in Figure 16. The Enterprise Gatekeeper is also neighbored with the TANDBERG Gatekeeper.

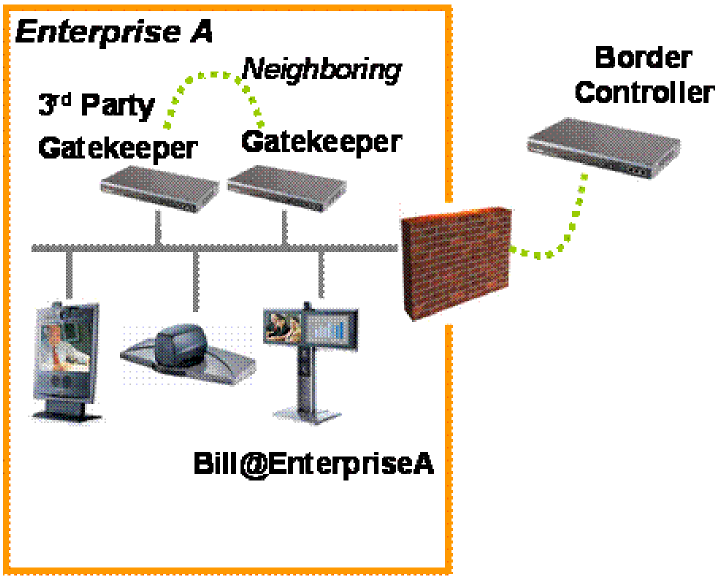



Figure 16: Neighboring with an enterprise gatekeeper

The TANDBERG Gatekeeper and Border Controller are configured as described in section 7.1, in order to provide firewall traversal.

## 7.3 Dialing Public IP addresses

Figure 17 The diagram above shows a private endpoint (1001) calling an endpoint on a public IP address. In this case the public endpoint is not registered to a Gatekeeper and can only be reached using its IP address. In order to successfully traverse the firewall it is necessary for the call to be relayed through the Border Controller: the TANDBERG Gatekeeper should not attempt to place the call directly to the public endpoint.

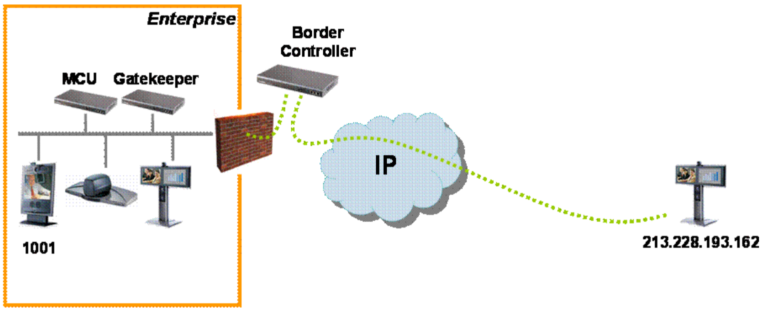



Figure 17: Dialing a public IP address

In order to achieve this:

- Within the Gatekeeper configuration — set "Calls to unknown IP addresses" to Indirect. This setting will force the Gatekeeper to forward calls to any IP address it does not have locally registered to the TANDBERG Border Controller, thereby allowing the Border Controller itself to relay the call to the endpoint on the public IP address.
- On the Border Controller, configure "Calls to unknown IP addresses" to *Direct*. This setting will allow the Border Controller to connect any call that it receives from the internal Gatekeeper out to systems on the public Internet.
- From Endpoint 1001, dial 213.228.193.162

## 7.4 Neighbored enterprises

If two sites have deployed Border Controllers for firewall traversal, the two Border Controllers may be neighbored to allow calls to be placed from one enterprise to another. Neighboring will reduce call setup time compared to URI dialing (described in section 6). The disadvantage of neighboring is that the Border Controllers have to be configured with each others addresses before the call can be made.

Gatekeeper and matching Border Controller are neighbored as described in section 7.1. Border Controller A and B are neighbored together, either with or without prefixes.

## 7.5 URI dialing from within the enterprise

This diagram shows a deployment to support URI dialing to other enterprises.

- Turn URI dialing OFF on the TANDBERG Gatekeeper. You want to use the Border Controller to resolve any H.323 URI received.
- Ensure that DNS Resolution Mode is turned on at the TANDBERG Border Controller. You want to use the Border Controller to resolve any H.323 URI received
- Configure the local domain name on both the Gatekeeper and the Border Controller.
- Configure the Border Controller with the address of a public DNS server.

- From an endpoint in enterprise A, dial the full H.323 URI. For example, Ben@EnterpriseB.com. Border Controller B is registered in DNS as responsible for enterprise B and will receive the incoming call and route it accordingly.

URI dialing will send all queries for a particular domain to the same Border Controller. If you want to have URI dialing covering multiple Border Controllers, nominate one as the master. That system is registered in DNS and is set up with all the other Border Controllers and Gatekeepers as neighbors. When the master receives a URI dialing request for an endpoint it does not know about, it will query its neighbors.

# 8 Call Policy

Your TANDBERG Border Controller allows you to set up policy to control which calls are allowed and even redirect selected calls to different destinations. You specify this policy by uploading a script written in the Call Processing Language (CPL). Each time a call is made the

Border Controller executes the script to decide, based on the source and destination of the call, whether to

- Proxy the call to its original destination
- Redirect the call to a different destination
- Reject the call.

The Border Controller will only execute scripts for source or destinations which are registered directly with the system.

The CPL script is uploaded via the Web interface under the *Border Controller Configuration → Files* web page.

The execution of the CPL script is controlled by the setting

```
xConfiguration Gatekeeper Policy Mode <On/Off>
```

Policy interacts with authentication (section 5.2). If authentication is enabled on the local Border Controller and a call is received from a remote, unauthenticated Gatekeeper, the call's source aliases will be removed from the call request before it is passed to the policy engine. This is because the unauthenticated source aliases could be forged and so should not be used for policy decisions in a secure environment.

The following sections give details of the Border Controller's implementation of the CPL language and should be read on conjunction with the CPL standard (RFC 3880[5]).

## 8.1 Making Decisions Based on Addresses

### 8.1.1 address-switch

The address-switch node allows the script to run different actions based on the source or destination aliases of the call. The address-switch specifies which fields to match and then a list of address nodes contains the possible matches and their associated actions.

The supported attributes on an address-switch and their interpretation are as follows:

**field**

| | |
|---|---|
| origin | Match against the source aliases. |
| destination | Match against the destination aliases. |
| original-destination | Match against the destination aliases. |

If the selected field contains multiple aliases then the Border Controller will attempt to match each address node with all of the aliases before proceeding to the next address node i.e. an address node matches if it matches any alias.

**subfield**

The following table gives the definition of subfields for each alias type, if a subfield is not specified for the alias type being matched then the not-present action will be taken.

| | |
|---|---|
| address-type | For all alias types the address-type subfield is the string `h323` |
| user | For URI aliases this selects the username part. For H.323 ID's it is the entire ID and for E.164 numbers it is the entire number. |
| host | For URI aliases this selects the domain name part. If the alias is an IP address then this subfield is the complete address in dotted decimal form. |
| port | For IP addresses this is the port number in decimal. |
| tel | For E.164 numbers this selects the entire string of digits. |
| alias-type | Gives a string representation of the type of alias. The type is inferred from the format of the alias. Possible types are:<br>*Address Type* — *Result*<br>URI — url-ID<br>H.323 ID — h323-ID<br>Dialed Digits — dialedDigits |
| display | Not defined for any alias types |

**address**

The address construct is used within an address-switch to specify addresses to match. Please note that all address comparisons ignore upper/lower case differences so `<address is="Fred">` will match "fred", "freD" etc.

| | |
|---|---|
| `is=`*string* | Selected field and subfield exactly match the given string. |
| `contains=`*string* | Selected field and subfield contain the given string. Note: The CPL standard only allows for this matching on the display subfield; however the Border Controller allows it on any type of field. |
| `subdomain-of=`*string* | If the selected field is numeric (e.g. the `tel` subfield) then this matches as a prefix; so `<address subdomain-of="555">` matches "5556734" etc. If the field is not numeric then normal domain name matching is applied; so `<address subdomain-of="company.com">` matches nodeA.company.com etc. |

**otherwise**

The otherwise node will be executed if the address specified in the address-switch was found but none of the preceding address nodes matched.

**not-present**

The not-present node is executed when the address specified in the address-switch was not present in the call setup message. This form is most useful when authentication is being used. With authentication enabled the Border Controller will only use authenticated aliases when running policy so the not-present action can be used to take appropriate action when a call is received from an unauthenticated user (see example in section 8.4).

## 8.2 CPL Script Actions

### 8.2.1 location

As the CPL script runs it maintains a list of addresses (H.323 IDs, URLs and E.164 numbers) which will be used as the destination of the call if a proxy node is executed. The location node allows the location set to be modified so that calls can be redirected to different destinations.

At the start of script execution the location set is initialized to empty for incoming calls and to the original destination for outgoing calls.

The following attributes are supported on location nodes

`Clear = "yes" | "no"`

Specifies whether to clear the current location set before adding the new location. The default is to append this location to the end of the set.

`url=`*string*

The new location to be added to the location set. The given string can specify a URL (user@domain.com), H.323 ID or an E.164 number.

### 8.2.2 proxy

On executing a proxy node the Border Controller will attempt to forward the call to the locations specified in the current location set. If multiple entries are in the location set then they are treated as different aliases for the same destination and are all placed in the destination alias field. If the current location set is empty the call will be forwarded to its original destination.

It is important to note that when a proxy node is executed script execution stops immediately i.e. there is currently no support for the proxy outputs `busy`, `noanswer` etc.

### 8.2.3 reject

If a reject node is executed the Border Controller stops any further script processing and rejects the current call.

## 8.3 Unsupported CPL Elements

The Border Controller does not currently support the following elements that are described in the CPL RFC. If an attempt is made to upload a script containing any of the following elements an error message will be generated and the Border Controller will continue to use its existing policy.

- time-switch
- string-switch
- language-switch
- time-switch
- priority-switch
- redirect
- mail
- log
- subaction
- lookup
- remove-location

## 8.4 CPL Examples

### 8.4.1 Call screening

Only allow calls from users with authenticated source addresses. See section 5.2 for details on how to enable authentication.

```
<cpl>
  <incoming>
    <address-switch field="origin">
      <not-present>
        <reject/>
      </not-present>
    </address-switch>
  </incoming>
</cpl>
```

### 8.4.2 Selective Call Screening

User ”fred” will not accept calls from anyone at ”annoying.com”, or from any unauthenticated users. All other users will allow any calls.

```
<cpl>
  <incoming>
    <address-switch field="destination">
```

```
      <address is="fred">
        <address-switch field="origin" subfield="host">
          <address subdomain-of="annoying.com">
            <reject/>
          </address>
          <otherwise>
            <proxy/>
          </otherwise>
          <not-present>
            <reject/>
          </not-present>
        </address-switch>
      </address>
    </address-switch>
  </incoming>
</cpl>
```

### 8.4.3 Call Redirection

Redirect all calls to user ”barney” to voicemail.

```
<cpl>
  <incoming>
    <address-switch field="destination">
      <address is="barney">
        <location clear="yes" url="barney@voicemail">
          <proxy/>
        </location>
      </address>
      <otherwise>
        <proxy/>
      </otherwise>
    </address-switch>
  </incoming>
</cpl>
```

# 9 Logging

The Border Controller provides logging for troubleshooting and auditing purposes.

The event log may be viewed from the command line by using the `eventlog` command, specifying the number of lines to display. Alternatively the web page *System Status* → *Event Log* may be used.

## 9.1 Controlling what is logged

You can control the verbosity with which the Border Controller logs information. All events have an associated level in the range [1-3]. Level 1 refers to high level events such as registration requests and call attempts. Level 2 events are recorded for incoming and outgoing message - H.323, LDAP etc excluding noisy messages such as H.460.18 keep-alives and H.245 video fast-updates. . Level 3 events include some of these noisy events. By default, logging is set to level 1.

## 9.2 Event log format

The event log is displayed in an extension of the UNIX syslog format:

```
date time host_name facility_name <PID>:  message_details
```

*date* and *time* represent the local time at which the message was logged. *host_name* is the name of the system generating the log message, *facility* — the name of the program generating the log message — will be `tandberg` for all messages originating from TANDBERG processes, but will differ for messages from third party processes which are used in the Border Controller product.

For all messages logged from the `tandberg` process the *message_details* field is structured to allow easy parsing. It consists of a number of human-readable *name=value* pairs, separated by a space. The first two fields are always:

| *Field* | *Example* | *Description* |
|---|---|---|
| Time | `Time=2006/20/01-14:02:17` | The UTC date and time at which the event was generated. |
| Event | `Event=RegistrationRequest` | The event which caused the log message to be generated. |

and the last field of the message is always the event level:

| *Field* | *Example* | *Description* |
|---|---|---|
| Level | `Level=1` | The level of the event being logged. |

## 9.3 Event Levels

Events are classified by importance as detailed in the table below. Level 1 is considered the most important. The system has a configured logging level. Events of level numerically equal to and lower than the configured logging level are recorded in the event log.

Table 1: Event levels

| Level | Description |
|---|---|
| Level 1 (User) | Easily human readable. Examples:<br>• call attempt/connected/disconnected<br>• registration attempt/accepted/rejected |
| Level 2 (Protocol) | Logs of protocol messages sent and received. |
| Level 3 (Protocol Verbose) | Protocol keepalives are suppressed at Level 2. At logging level 3, keepalives are also logged. |

## 9.4 Logged Events

The Events logged as are follows:

Table 2: Events logged at level 1

| Event | Description |
|---|---|
| Eventlog Cleared | An operator cleared the event log |
| Admin Session Start | An administrator has logged onto the system |
| Admin Session Finish | An administrator has logged off the system |
| System Configuration Changed | An item of configuration on the system has changed. The detail event parameter contains the name of the changed configuration item and its new value. |
| Policy Change | A policy file has been updated |
| Registration Requested | A registration has been requested |
| Registration Accepted | A registration request has been accepted |
| Registration Rejected | A registration request has been rejected. The Reason event parameter contains the H225 cause code. Optionally, the Detail event parameter may contain a textual representation of the H.225 additional cause code. |

Table 2: Level 1 Events (continued)

| Event | Description |
| --- | --- |
| Registration Removed | A registration has been removed by the gatekeeper/border controller. The Reason event parameter specifies the reason why the registration was removed. This is one of:<br>• Authentication change<br>• Conflicting zones<br>• Operator forced removal<br>• Operator forced removal (all registrations removed) |
| Call Answer Attempted | An attempt to answer a call has been made |
| Call Attempted | A call has been attempted. |
| Call Connected | A call has been connected |
| Call Disconnected | A call has been disconnected |
| Call Rejected | A call has been rejected. The Reason event parameter contains a textual representation of the H.225 additional cause code. |
| Call Bandwidth Changed | The bandwidth of a call has changed. |
| External Server Communication Failure | Communication with an external server failed unexpectedly. The event detail data should differentiate between 'no response' and 'request rejected' (i.e. NACK rather than silence) Servers concerned are:<br>• DNS<br>• LDAP servers<br>• Neighbour Gatekeeper<br>• NTP servers |
| System Start | The operating system has started. |
| System Shutdown | The operating system was shutdown. |
| Application Start | The Border Controller has started. Further detail may be provided in the event data 'detail' field. |
| Application Failed | The Border Controller application is out of service due to an unexpected failure |
| License Limit Reached | Licensing limits for a given feature have been reached. The event detail field specifies the facility/limits concerned. Possible values for the detail field are:<br>• Non Traversal Call Limit Reached<br>• Traversal Call Limit Reached |

Table 3: Events logged at level 2

| Event | Description |
|---|---|
| Incoming Message | An incoming message has been received |
| Outgoing Message | An outgoing message has been sent |

### 9.4.1 Event data

Each Event will have associated data fields. Fields are listed below in the order in which they appear in the log message.

Table 4: Event data

| Field | Description | Applicable events |
|---|---|---|
| Protocol | Specifies which protocol was used for the communication. Valid values are TCP or UDP | • Call Attempted<br>• Call Bandwidth Changed<br>• Call Connected<br>• Call Disconnected<br>• Call Rejected<br>• External Server Communication Failure<br>• Incoming Message<br>• Outgoing Message<br>• Policy Change<br>• Registration Accepted<br>• Registration Rejected<br>• Registration Removed<br>• Registration Requested |
| Reason | Textual string containing any reason information associated with an event. | • Call Rejected<br>• External Server Communication Failure<br>• Registration Rejected<br>• Registration Removed |
| Service | Specifies which protocol was used for the communication. A service entry is one of H.225, H.245,NTP,DNS,LDAP, Neighbour Gatekeeper | • External Server Communication Failure<br>• Incoming Message<br>• Outgoing Message |
| Message Type | Specifies the type of the message. | • Incoming Message<br>• Outgoing Message |

Table 4: Event data (continued)

| Field | Description | Applicable events |
|---|---|---|
| Src-ip | Specifies the source IP address (the IP address of the device attempting to establish communications). The source IP is recorded in the dotted decimal format: (number).(number).(number).(number) or the IPv6 colon separated format. | • Call Attempted<br>• Call Bandwidth Changed<br>• Call Connected<br>• Call Disconnected<br>• Call Rejected<br>• External Server Communication Failure<br>• Incoming Message<br>• Outgoing Message<br>• Policy Change<br>• Registration Accepted<br>• Registration Rejected<br>• Registration Removed<br>• Registration Requested |
| Dst-ip | Specifies the destination IP address (the IP address of the destination for a communication attempt). The destination IP is recorded in the same format as Src-ip. | As Src-ip |
| Dst-port | Specifies the destination port: the IP port of the destination for a communication attempt | As Src-ip |
| Src-port | Specifies the source port: the IP port of the device attempting to establish communications. | As Src-ip |
| Src-Alias | • If present, the first H.323 Alias associated with the originator of the message<br>• If present, the first E.164 Alias associated with the originator of the message | • Registration Requested<br>• Call Attempted<br>• Call Connected<br>• Call Disconnected<br>• Call Rejected<br>• Call Bandwidth Changed<br>• Incoming Message[1]<br>• Outgoing Message[1] |

[1] Included if event parameter relevant or available for message concerned.

Table 4: Event data (continued)

| Field | Description | Applicable events |
|---|---|---|
| Dst-Alias | • If present, the first H.323 Alias associated with the recipient of the message<br>• If present, the first E.164 Alias associated with the recipient of the message | • Registration Accepted<br>• Registration Removed<br>• Registration Rejected<br>• Call Attempted<br>• Call Connected<br>• Call Disconnected<br>• Call Rejected<br>• Incoming Message[1]<br>• Outgoing Message[1]<br>• Call Bandwidth Changed |
| Time | A full UTC timestamp in YYYY/MM/DD-HH:MM:SS format. Using this format permits simple ASCII text sorting/ordering to naturally sort by time. This is included due to the limitations of standard syslog timestamps. | All Events |
| Level | The level of the event as defined in section 9.3. All events | |

## 9.5 Remote Logging

It is often convenient to collect all event logs in a single location. A computer running a BSD style syslog server, as defined in RFC3164[4] , may be used as the central log server — ensure that remote logging is enabled. A Border Controller will not act as a central logging server for other systems.

The Border Controller should be configured with the address of the central log server:

```
xConfiguration Log Server Address: server_address
```

# 10 Software Upgrade

Software upgrade can be done in one of two ways:

1. Using a web browser (HTTP/HTTPS).
2. Using secure copy (SCP).

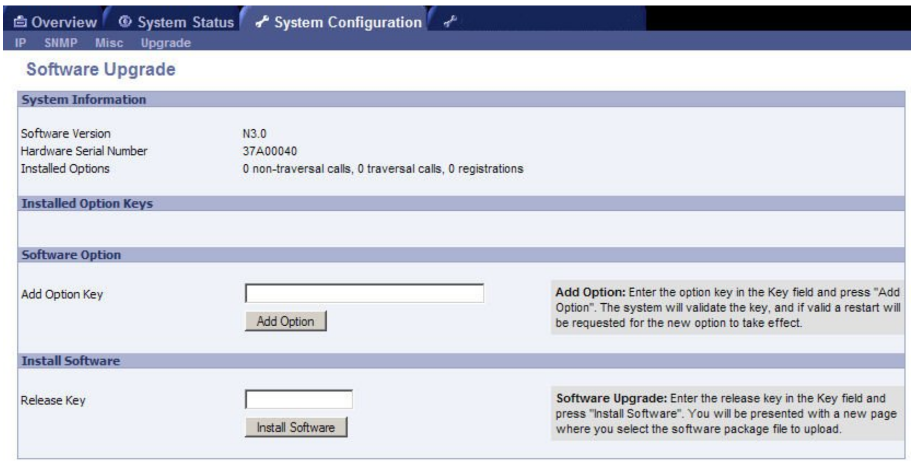
**NOTE** To upgrade the Border Controller, a valid Release key and software file is required. Contact your TANDBERG representative for more information.

**NOTE** Configuration is restored after performing an upgrade but we recommend that you make a backup of the existing configuration using the TANDBERG Management Suite before performing the upgrade.

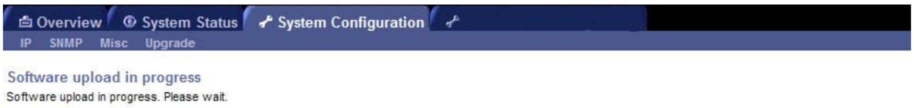
## 10.1 Upgrading Using HTTP(S)

To upgrade using HTTP(S), do the following:

- Point your browser at the IP address of the Border Controller. You will be prompted for your user name and password.
- Enter `admin` as the user name and enter the password, then press OK.
- Select the System Configuration tab, and the upgrade section.
- Enter the release key and press Install Software. You will get a new screen where you can upload the software image:



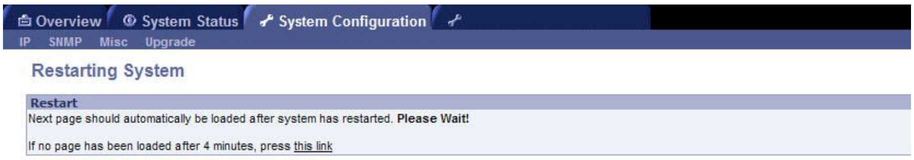

- Browse to the file containing the software and press Install. You should see a page indicating that upload is in progress:




- When the upload is completed you should see the following:




- Press Restart. You should see a confirmation window:




- The system will then perform a second reboot to restore system parameters. After 3–4 minutes, the Border Controller is ready for use.

## 10.2 Upgrading Using SCP

Using SCP you need to transfer two files to the Border Controller:

1. A text file containing the release key.
2. A file containing the software image.

**NOTE** Make sure you transfer the release key file before transferring the software image. Also make sure you name the files exactly as described below.

**NOTE** The release key file should contain just the 16 character release key.

To upgrade using SCP, do the following:

- Make sure the system is turned on and available on IP.
- Upload the release key file using scp to the /tmp folder on the system e.g.

  scp release-key root@10.47.8.247:/tmp/release-key

- Enter password when prompted.
- Copy the software image using SCP. The target name must be `/tmp/tandberg-image.tar.gz`, e.g.

  `scp s42100q30.tar.gz root@10.47.8.247:/tmp/tandberg-image.tar.gz`

- Enter password when prompted.
- Wait until the software has installed completely. This should not take more than two minutes.
- Reboot the system. After about four minutes the system will be ready to use.

# 11 Command Reference

This chapter lists the basic usage of each command. The commands also support more advanced usage, which is outside the scope of this document.

## 11.1 Status

The status root command, `xstatus`, returns status information from the Border Controller.

To list all status information, type:

```
xstatus
```

Status is reported hierarchically beneath the status root. It is possible to reduce the amount of information returned by xstatus by specifying a more detailed status command. To list all xstatus commands available at the root level type:

```
xstatus ?
```

### 11.1.1 calls

```
xstatus Calls
xstatus Calls Call n
```

Returns a list of active calls on the system or information about a specific call.

### 11.1.2 ethernet

```
xstatus Ethernet
xstatus Ethernet MacAddress
xstatus Ethernet Speed
```

Reports the currently active configuration of the Ethernet interface.

| | |
|---|---|
| MacAddress | The MAC address of the LAN 1 interface. |
| Speed | The speed of the Ethernet link. Reports Down if the link is down or not connected. |

### 11.1.3 externalmanager

```
xstatus ExternalManager
```

Returns information about the external manager. The External Manager is the remote system (such as the Tandberg Management System (TMS)) used to manage the endpoints and network infrastructure.

| | |
|---|---|
| Address | IP address of the external manager. |
| Protocol | Protocol used to communicate with the external manager. |
| URL | URL used to communicate with the external manager. |

### 11.1.4 feedback

```
xstatus Feedback
xstatus Feedback n
```

Returns all currently registered feedback expressions or the feedback expression at index $n$.

### 11.1.5 IP

```
xstatus IP
```

Returns the active IP configuration of the system with IP address, subnet mask and gateway.

If you have changed the IP configuration without rebooting, xstatus IP will return the original settings currently in effect.

| | |
|---|---|
| Address | IP address. |
| SubnetMask | IP subnet mask. |
| Gateway | Default gateway. |
| DNS Server | The DNS servers in use |

### 11.1.6 LDAP

```
xstatus LDAP
```

Reports the status of any connection to an LDAP server.

### 11.1.7 Links

```
xstatus Links
xstatus Links Link n
```

Reports call and bandwidth information for all links on the system.

| | |
|---|---|
| Name | Name assigned to this link. |
| Calls | A list of call indices for calls currently active on this link. |
| Bandwidth | Total and per call bandwidth limits on this link, together with bandwidth currently in use. |

### 11.1.8 NTP

`xstatus NTP`

Reports the status of any connection to an NTP server.

### 11.1.9 Pipes

```
xstatus Pipes
xstatus Pipes Pipe n
```

Reports call and bandwidth information for all pipes on the system.

### 11.1.10 Registrations

```
xstatus Registrations
xstatus Registrations Registration n
```

Returns a list of registered endpoints on the system or information about a specific registration.

### 11.1.11 ResourceUsage

`xstatus ResourceUsage`

Reports information about the usage of system resources.

| | |
|---|---|
| PortRegistrations | Total number of currently registered endpoints and services. See glossary for definition. |
| MaxPortRegistrations | Maximum number of registered endpoints and services since system start. |
| TraversalCalls | Number of currently active traversal calls. |
| MaxTraversalCalls | Maximum number of traversal calls since system start. |
| TotalTraversalCalls | Total number of traversal calls since system start. |
| NonTraversalCalls | Number of currently active non traversal calls. |
| MaxNonTraversalCalls | Maximum number of non traversal calls since system start. |
| TotalNonTraversalCalls | Total number of non traversal calls since system start. |

### 11.1.12 SubZones

`xstatus SubZones`

Reports call and bandwidth information for all subzones on the system.

#### 11.1.13 SystemUnit

`xstatus SystemUnit`

Reports information about the system as follows:

Product name
Uptime
Software version
Software name
Release date
Number of calls supported
Number of registered endpoints and services supported
Hardware serial number

#### 11.1.14 Zones

`xstatus Zones`

Reports the call and bandwidth information for all zones on the system. Also shows status of the zone as a whole and the status of each gatekeeper in the zone.

### 11.2 Configuration

The configuration root command, `xconfiguration`, is used to set configuration settings.

To list all xconfiguration commands type:

`xconfiguration ?`

To list all configuration data, type:

`xconfiguration`

To show a specific configuration value, type:

`xconfiguration` *name*

To show usage information for a specific configuration value, type:

`xconfiguration` *name* `?`

To set a configuration element type:

`xconfiguration` *name param1*`:` *value1 param2*`:` *value2*

**NOTE** Remember to use the colon after naming the parameters.

#### 11.2.1 Authentication

Configuration parameters relating to how an endpoint authenticates itself with the Border Controller.

`xConfiguration Authentication Credential [1..1000] Name: <username>`

Specifies the username of a credential in the local authentication database.

`xConfiguration Authentication Credential [1..1000] Password: <password>`

Specifies the password of a credential in the local authentication database

`xconfiguration Authentication Database: <LocalDatabase/LDAPDatabase>`

Select between a local database and a remote LDAP repository for the storage of password information for authentication. The default is `LocalDatabase`.

`xConfiguration Authentication LDAP BaseDN: <S: 0, 255>`

The Distinguished Name to use when connecting to an LDAP server. The default is an empty string.

`xConfiguration Authentication Mode: <On/Off>`

Whether or not to use H.235 authentication of calls and registrations. The default is `Off` — no authentication is required.

### 11.2.2 Ethernet

`xConfiguration Ethernet Speed: <Auto/10half/10full/100half/100full>`

Sets the speed of the Ethernet link. Use auto to automatically configure the speed. To get the current speed, use `xstatus Ethernet Speed`. You must restart the system for changes to take effect. The default is `Auto`.

### 11.2.3 ExternalManager

`xConfiguration ExternalManager Address: <IPAddr>`

Sets the IP address of the External Manager. The External Manager is the remote system(such as the TANDBERG Management System (TMS)) used to manage endpoints and network infrastructure.

`xConfiguration ExternalManager Path: <path>`

Sets the URL of the External Manager.

### 11.2.4 Gatekeeper

Commands under the `Gatekeeper` node control aspects of the systems operation relating to its operation as an H.323 gatekeeper.

`xConfiguration Gatekeeper Alternates Monitor: <On/Off>`

Controls whether or not alternate gatekeepers are periodically interrogated to ensure that they are still functioning. Non-functional alternates will not receive Location Requests in order to prevent delays during call setup.

`xConfiguration Gatekeeper Alternates Alternate [1..5] Address: <IPAddr>`

Set the IP address of an alternate Border Controller. Up to 5 alternates may be configured. When the Border Controller receives a Location Request, all alternates will also be queried.

`xConfiguration Gatekeeper Alternates Alternate [1..5] Port: <IPAddr>`

Set the IP port of an alternate Border Controller. The default is `1719`.

`xConfiguration Gatekeeper AutoDiscovery: <On/Off>`

Specifies whether or not the Border Controller responds to gatekeeper discovery requests from endpoints. The default is `On`.

`xConfiguration Gatekeeper CallsToUnknownIPAddresses: <Off/Direct/Indirect>`

Specifies whether or not the Border Controller will attempt to call systems which are not registered with it or one of its neighbor gatekeepers. If `Off` is selected, the Border Controller will not allow calls to be made to unknown IP addresses. If `Indirect` is selected the Border Controller will not place the call itself, instead it will LRQ its neighbor systems so that they may place the call. If `Direct` is selected, the Border Controller will attempt to dial the IP address. See 3.5 for further detail. The default is `Indirect`.

`xConfiguration Gatekeeper CallTimeToLive: <60..65534>`

Interval in seconds at which endpoints are polled to verify that they are still in a call. The default is `120` seconds.

`xConfiguration Gatekeeper DNSResolution Mode: <On/Off>`

Determines whether or not DNS lookup of H.323 URI's is enabled on this system. The default is `On`.

`xConfiguration Gatekeeper Downspeed PerCall Mode: <On/Off>`

Determines whether or not the system will attempt to down-speed a call if there is insufficient per-call bandwidth configured to fulfill the request. The default is `On`.

`xConfiguration Gatekeeper Downspeed Total Mode: <On/Off>`

Determines whether or not the system will attempt to down-speed a call if there is insufficient total bandwidth available to fulfill the request. The default is `On`.

`xConfiguration Gatekeeper ForwardLocationRequests: <On/Off>`

Determines behavior on receipt of a location request (LRQ) from another Gatekeeper. If set to *on*, the Border Controller will first try to resolve the request locally. If it cannot, the request will be forwarded to neighbor Gatekeepers. The default is `On`.

`xConfiguration Gatekeeper LocalDomain DomainName`

DNS name of the domain that the Gatekeeper is responsible for. Used when searching for matching endpoint registrations.

`xConfiguration Gatekeeper LocalPrefix: <prefix>`

Set the local zone prefix of the system.

`xConfiguration Gatekeeper Policy Mode: <On/Off>`

Determines whether or not the CPL policy engine is active. The default is `On`.

`xConfiguration Gatekeeper Registration AllowList [1..1000] Pattern: <pattern>`

Specifies a pattern in the registration allowed list. If one of an endpoint's aliases matches one of the patterns in the AllowList, the registration will be allowed.

`xConfiguration Gatekeeper Registration DenyList [1..1000] Pattern: <pattern>`

Specifies a pattern in the registration denied list. If one of an endpoint's aliases matches one of the patterns in the DenyList the registration will be denied.

`xConfiguration Gatekeeper Registration RestrictionPolicy: <None/AllowList/DenyList>`

Policy in use to determine who may register with the system. The default is `None`.

`xConfiguration Gatekeeper TimeToLive: <60..65534>`

The interval at which the system polls the endpoint in order to verify that it is still functioning. Specified in seconds. The default is `1800` seconds.

### 11.2.5 HTTP/HTTPS

Command under the `HTTP` and `HTTPS` nodes control web access to the Border Controller.

`xConfiguration HTTP Mode: <On/Off>`

Enables/disables HTTP support. You must restart the system for changes to take effect. The default is `On`.

`xConfiguration HTTPS Mode: <On/Off>`

Enables/disables HTTPS support. You must restart the system for changes to take effect. The default is `On`. If web access is required, you are recommended to enable HTTPS and disable HTTP for improved security.

### 11.2.6 IP

Configuration of IP related parameters. The TANDBERG Border Controller may be configured to use either IPv4 or IPv6. When entering IPv4 addresses, dotted quad notation is used: 127.0.0.1, when using IPv6 addresses are entered in colon hexadecimal form: FE80::2AA:FF:FE9A:4CA2.

`xConfiguration IPProtocol: <IPv4/IPv6>`

Selects whether the Border Controller is operating in IPv4 or IPv6 mode.

`xConfiguration IP Address: <IPAddr>`

The IPv4 address of the system.

`xConfiguration IP SubnetMask: <IPAddr>`

The IPv4 subnet mask of the system.

`xConfiguration IP Gateway: <IPAddr>`

The IPv4 gateway of the system.

`xConfiguration IP V6 Address: <IPAddr>`

The IPv6 address of the system.

`xConfiguration IP V6 Gateway: <IPAddr>`

The IPv6 gateway of the system.

All the IP commands listed above require a system restart before they take effect.

`xConfiguration IP DNS Server [1..5] Address: <IPAddr>`

Sets the IP address of the DNS servers to be used when resolving domain names. Normally only the first DNS server will be queried for address resolution. If it fails to respond, all DNS servers will be queried. You must restart the system for changes to take effect.

`xConfiguration IP DNS Domain Name: <name>`

When attempting to resolve a domain name which is not fully qualified, *name* will be appended to the domain name before the query to the DNS server is executed.

This parameter is only used when attempting to resolve server addresses such as LDAP servers, NTP servers etc. It plays no part in URI dialing: see `xconfiguration gatekeeper localdomain`

### 11.2.7 LDAP

Parameters under the `LDAP` node control the Border Controller's communication with an LDAP server.

`xConfiguration LDAP Encryption: <Off/TLS>`

Sets the encryption mode to be used on the connection to the LDAP server. The default is `Off`.

`xConfiguration LDAP Password: <password>`

Sets the password to be used when binding to the LDAP server.

`xConfiguration LDAP Server Address: <IPAddr>`

Sets the IP address of the LDAP server to be used when making LDAP queries.

`xConfiguration LDAP Server Port: <1..65534>`

Sets the IP port of the LDAP server to be used when making LDAP queries.

`xConfiguration LDAP UserDN: <userdn>`

Sets the user DN to be used when binding to the LDAP server.

### 11.2.8 Links

`xConfiguration Links Link [1..100] Name: <linkname>`

Specifies the name of a link in the list of links.

`xConfiguration Links Link [1..100] Node1 Name: <nodename>`

Specifies the first node of a link. A node name may be either a Zone name or a SubZone name.

`xConfiguration Links Link [1..100] Node2 Name: <nodename>`

Specifies the second node of a link. A node name may be either a Zone name or a SubZone name.

`xConfiguration Links Link [1..100] Pipe1 Name: <pipename>`

First pipe associated with a link.

`xConfiguration Links Link [1..100] Pipe2 Name: <pipename>`

Second pipe associated with a link.

`xConfiguration Links TraversalLink Pipe1 Name: <pipename>`

First pipe associated with the traversal link.

`xConfiguration Links TraversalLink Pipe2 Name: <pipename>`

Second pipe associated with the traversal link.

### 11.2.9 Log

`xConfiguration Log Level: <1..3>`

Controls the granularity of event logging with `1` being the least verbose, `3` the most.

### 11.2.10 NTP

`xConfiguration NTP Address: <IPAddr>`

Sets the IP address of the NTP server to be used when synchronizing system time. Accurate timestamps play an important part in authentication, helping to guard against replay attacks.

### 11.2.11 Option Key

`xConfiguration Option [1..64] Key: <optionkey>`

Specify the option key of your software options.

`xstatus system software configuration` can be used to discover the existing options. You must restart the system for changes to take effect.

### 11.2.12 Pipes

`xConfiguration Pipes Pipe [1..100] Bandwidth Total Limit: <1..100000000>`

Bandwidth associated with a pipe, keyed by index.

`xConfiguration Pipes Pipe [1..100] Bandwidth Total Mode: <None/Limited/Unlimited>`

Whether or not a given pipe is enforcing total bandwidth restrictions. `None` corresponds to no bandwidth available.

`xConfiguration Pipes Pipe [1..100] Bandwidth PerCall Limit: <1..100000000>`

Per call bandwidth of a pipe.

`xConfiguration Pipes Pipe [1..100] Bandwidth PerCall Mode: <None/Limited/Unlimited>`

Whether or not a given pipe is enforcing per-call bandwidth restrictions. `None` corresponds to no bandwidth available.

`xConfiguration Pipes Pipe [1..100] Name: <pipename>`

Name for a pipe.

### 11.2.13 Session

`xConfiguration Session TimeOut: <0..65534>`

Controls how long an adminstration session (HTTPS, Telnet or SSH) may be inactive before the session is timed out. A value of `0` turns session time outs off. The default is `0`. You must restart the system for changes to take effect.

### 11.2.14 SNMP

`xConfiguration SNMP CommunityName: <name>`

SNMP Community names are used to authenticate SNMP requests. SNMP requests must have this ‘password’ in order to receive a response from the SNMP agent in the Gatekeeper. You must restart the system for changes to take effect.

`xConfiguration SNMP Mode: <On/Off>`

Turn on/off SNMP support. You must restart the system for changes to take effect.

`xConfiguration SNMP SystemContact: <name>`

Used to identify the system contact via SNMP tools such as TANDBERG Management Suite or HPOpenView. You must restart the system for changes to take effect.

`xConfiguration SNMP SystemLocation: <name>`

Used to identify the system location via SNMP tools such as TANDBERG Management Suite or HPOpenView. You must restart the system for changes to take effect.

### 11.2.15 SSH

`xConfiguration SSH Mode: <On/Off>`

Enables/disables SSH and SCP support. You must restart the system for changes to take effect.

### 11.2.16 Subzones

`xConfiguration SubZones DefaultSubZone Bandwidth PerCall Limit: <1..100000000>`

Per call bandwidth of the default subzone.

`xConfiguration SubZones DefaultSubZone Bandwidth PerCall Mode: <None/Limited/Unlimited>`

Whether or not the default subzone is enforcing total bandwidth restrictions. `None` corresponds to no bandwidth available.

`xConfiguration SubZones DefaultSubZone Bandwidth Total Limit: <1..100000000>`

Total bandwidth available on the default subzone.

`xConfiguration SubZones DefaultSubZone Bandwidth Total Mode: <None/Limited/Unlimited>`

Whether or not the default subzone is enforcing per-call bandwidth restrictions. `None` corresponds to no bandwidth available.

`xConfiguration SubZones TraversalSubZone Bandwidth PerCall Limit: <1..100000000>`

Per-call bandwidth available on the traversal subzone.

`xConfiguration SubZones TraversalSubZone Bandwidth PerCall Mode: <None/Limited/Unlimited>`

Whether or not the traversal subzone is enforcing per-call bandwidth restrictions. `None` corresponds to no bandwidth available.

`xConfiguration SubZones TraversalSubZone Bandwidth Total Limit: <1..100000000>`

Total bandwidth available on the traversal subzone.

`xConfiguration SubZones TraversalSubZone Bandwidth Total Mode: <None/Limited/Unlimited>`

Whether or not the traversal subzone is enforcing total bandwidth restrictions. `None` corresponds to no bandwidth available.

`xConfiguration SubZones SubZone [1..100] Bandwidth PerCall Limit: <1..100000000>`

Per-call bandwidth available on the indexed subzone.

`xConfiguration SubZones SubZone [1..100] Bandwidth PerCall Mode: <None/Limited/Unlimited>`

Whether or not the indexed subzone is enforcing per-call bandwidth restrictions. `None` corresponds to no bandwidth available.

`xConfiguration SubZones SubZone [1..100] Bandwidth Total Limit: <1..100000000>`

Total bandwidth available on the indexed subzone.

`xConfiguration SubZones SubZone [1..100] Bandwidth Total Mode: <None/Limited/Unlimited>`
Whether or not the indexed subzone is enforcing total bandwidth restrictions. `None` corresponds to no bandwidth available.

`xConfiguration SubZones SubZone [1..100] Name: <subzonename>`

Name of the indexed subzone.

`xConfiguration SubZones SubZone [1..100] Subnet IP Address: <IPAddr>`

IP to match an endpoint which belongs in this subzone.

`xConfiguration SubZones SubZone [1..100] Subnet IP PrefixLength: <IPAddr>`

Number of bits which must match for an IP address to belong in this subzone.

### 11.2.17 SystemUnit

`xConfiguration SystemUnit Name: <name>`

The name of the unit. Choose a name that uniquely identifies the system.

`xConfiguration SystemUnit Password: <password>`

Specify the password of the unit. The password is used to login with Telnet, HTTP(S), SSH, SCP, and on the serial port. To set an empty password type

`xConfiguration SystemUnit Password: ""`

### 11.2.18 Telnet

`xConfiguration Telnet Mode: <On/Off>`

Enables/disables Telnet support. For secure operation you should use ssh in preference to telnet. You must restart the system for changes to take effect.

### 11.2.19 TimeZone

`xConfiguration TimeZone Name: <timezone name>`

Sets the local timezone. Timezone names follow the POSIX naming convention e.g. Europe/London or America/New_York.

### 11.2.20 Traversal

`xConfiguration Traversal UDPProbe RetryInterval: <seconds>`

Interval with which a failed attempt to establish a UDP channel should be repeated.

`xConfiguration Traversal UDPProbe RetryCount: <count>`

Number of attempts at re-establishing a failed UDP channel.

xConfiguration Traversal UDPProbe KeepAliveInterval: <seconds>

Interval with which a UDP channel should be refreshed. xConfiguration Traversal TCPProbe RetryInterval: <seconds >

Interval with which a failed attempt to establish a TCP channel should be repeated.

xConfiguration Traversal TCPProbe RetryCount: <count>

Number of attempts at re-establishing a failed TCP channel.

xConfiguration Traversal TCPProbe KeepAliveInterval: <seconds>

Interval with which a TCP channel should be refreshed.

xConfiguration Traversal Media RTP Port: <port>

UDP port to which media should be sent. Conventionally this will be an even numbered port. The default is 2776.

xConfiguration Traversal Media RTCP Port: <port>

UDP port to which media control information should be sent. Conventionally this will be set to RTP port + 1. The default is 2777.

### 11.2.21 Zones

Traversal zones control how the Border Controller communicates with a Gatekeeper which it is cooperating with to provide firewall traversal.

xConfiguration Zones TraversalZone [1..50] Name: <name>

Sets the name of the TANDBERG Gatekeeper which is allowed to connect to this Border Controller.

xConfiguration Zones TraversalZone [1..100] HopCount: <count>

Specifies the hop count to be used when originating an LRQ.

xConfiguration Zones TraversalZone [1..100] Match [1..5] Mode: <AlwaysMatch/PatternMatch/Dis

The prefix match mode determines when an LRQ will be sent to gatekeepers in the zone. If the mode is set to AlwaysMatch the zone will always be queried. If the mode is set to PatternMatch, the zone will only be queried if the alias queried for matches the corresponding pattern. If the mode is set to Disabled the zone will never be queried.

xConfiguration Zones TraversalZone [1..100] Match [1..5] Pattern String: <pattern>

The pattern to be used when deciding whether or not to query a zone. This is only used if the zone's match mode is set to AlwaysMatch.

xConfiguration Zones TraversalZone [1..100] Match [1..5] Pattern Type: <Prefix/Suffix>

Determines whether the pattern string being checked should appear at the beginning or end of an alias.

`xConfiguration Zones TraversalZone [1..100] Match [1..5] Pattern Behaviour: <Strip/Leave>`

Determines whether the matched pattern should be removed from the alias before an LRQ is sent to the indicated zone.

`xConfiguration Zones Zone [1..100] Name: <name>`

An administrator specified name for the zone.

`xConfiguration Zones Zone [1..100] Gatekeeper [1..6] Address: <address>`

Specifies the IP addresses of the gatekeepers in the zone. Multiple addresses allows support for alternate gatekeepers.

`xConfiguration Zones Zone [1..100] Gatekeeper [1..6] Port: <port>`

Specifies the port on which the indexed gatekeeper is listening for RAS messages.

`xConfiguration Zones Zone [1..100] HopCount: <count>`

Specifies the hop count to be used when originating an LRQ.

`xConfiguration Zones Zone [1..100] Monitor: <On/Off>`

If zone monitoring is enabled, an LRQ will be periodically sent to the zone gatekeeper. If it fails to respond, that gatekeeper will be marked as inactive.

`xConfiguration Zones Zone [1..100] Match [1..5] Mode: <AlwaysMatch/PatternMatch/Disabled>`

The zone match mode determines when an LRQ will be sent to gatekeepers in the zone. If the mode is set to `AlwaysMatch` the zone will always be queried. If the mode is set to `PatternMatch`, the zone will only be queried if the alias queried for matches the corresponding pattern. If the mode is set to `Disabled` the zone will never be queried.

`xConfiguration Zones Zone [1..100] Match [1..5] Pattern String: <pattern>`

The pattern to be used when deciding whether or not to query a zone. This is only used if the zone's match mode is set to `AlwaysMatch`.

`xConfiguration Zones Zone [1..100] Match [1..5] Pattern Type: <Prefix/Suffix>`

Determines whether the pattern string being checked should appear at the beginning or end of an alias.

`xConfiguration Zones Zone [1..100] Match [1..5] Pattern Behaviour: <Strip/Leave>`

Determines whether the matched pattern should be removed from the alias before an LRQ is sent to the indicated zone.

## 11.3 Command

The command root command, `xcommand`, is used to execute commands on the Border Controller.

To list all xcommands type

```
xcommand ?
```

To get usage information for a specific command, type

```
xcommand <commandname> ?
```

### 11.3.1 AllowListAdd

```
xCommand AllowListAdd <allowed_alias>
```

Adds an entry to the allow list, used by the registration restriction policy.

### 11.3.2 AllowListDelete

```
xCommand AllowListDelete <index>
```

Removes the pattern from the allow list at the specified index.

### 11.3.3 Boot

```
xCommand Boot
```

Reboots the Border Controller. This takes approximately 2 minutes to complete.

### 11.3.4 CheckBandwidth

```
xCommand CheckBandwidth <node1> <node2> <bandwidth> <calltype>
```

Diagnostic function for verifying bandwidth control. Node1, Node2 are the case sensitive names of the nodes, bandwidth the required bandwidth and calltype one of Traversal or NonTraversal.

### 11.3.5 CredentialAdd

```
xCommand CredentialAdd <username> <password>
```

Adds the given username and password to the local authentication database.

### 11.3.6 CredentialDelete

```
xCommand CredentialDelete <index>
```

Deletes the indexed credential.

### 11.3.7 DefaultLinksAdd

```
xCommand DefaultLinksAdd
```

Restores the factory default links for bandwidth control.

### 11.3.8 DefaultValuesSet

```
xCommand DefaultValuesSet Level <level>
```

Resets system parameters to default values. Level 1 will reset most parameters. There are currently no level 2 parameters, so setting that level has the same effect as setting level 1. Level 3 resets all level 1 and 2 parameters as well as the following:

- IP address, subnet mask, gateway and interface speed. The default IP address is 192.168.0.100.
- COM port baud rate, speed, data bits, parity, stop bits
- SNMP community name and host address
- system name
- password
- option key
- release key

Note that DefaltValuesSet will not add the links with which the system ships from the factory. Use the `DefaultLinksAdd` command to do that. Certificates and policy files are not removed

### 11.3.9 DenyListAdd

```
xCommand DenyListAdd <denied_alias>
```

Add an entry to the deny list. This is used by the registration restriction policy.

### 11.3.10 DenyListDelete

```
xCommand DenyListDelete <index>
```

Removes the pattern from the deny list at the specified index.

### 11.3.11 DisconnectCall

```
xCommand DisconnectCall <callid>
```

Disconnects the specified call.

### 11.3.12 FeedbackRegister

```
xCommand FeedbackRegister <ID> <URL> <Expression>
```

Registers for notifications on the event or status change described by the Expression. Notifications are sent in XML format to the specified URL. Up to 15 Expressions may be registered for each of 3 feedback IDs.

The following Expressions are valid:

Event, Event/CallAttempt, Event/Connected, Event/Disconnected, Event/ConnectionFailure, Event/Registration, Event/Unregistration, Event/Bandwidth, Status, Status/Calls, Status/Registrations, History, History/Calls, History/Registrations

The following would be a typical use: (Back slashes are used to indicate continuation lines)

```
xCommand FeedbackRegister ID:1
URL:http://10.1.1.1/SystemManagementService.asmx
Expression:Event/CallAttempt,Status/Registration FeedbackDeregister
```

### 11.3.13 FeedbackDeregister

```
xCommand FeedbackDeregister <ID>
```

Deregisters the specified Feedback Expression. All registered Feedback Expressions may be removed with `xCommand FeedbackDeregister 0 LinkAdd`

### 11.3.14 FindRegistration

```
xCommand FindRegistration <alias>
```

Returns information about the registartion associated with the given alias.

### 11.3.15 LinkAdd

```
xCommand LinkAdd <linkname> <node1> <node2> <pipe1> <pipe2>
```

Adds a new link to the link list.

### 11.3.16 LinkDelete

```
xCommand LinkDelete <index>
```

Deletes the indexed link.

### 11.3.17 OptionKeyAdd

```
xCommand OptionKeyAdd <key>
```

Adds a new option key.

### 11.3.18 OptionKeyDelete

```
xCommand OptionKeyDelete <index>
```

Deletes the indexed option key.

### 11.3.19 PipeAdd

```
xCommand PipeAdd <name> <totalmode> <total> <percallmode> <percall>
```

Adds and configures a new pipe.

### 11.3.20 PipeDelete

```
xCommand PipeDelete <index>
```

Deletes the indexed pipe.

### 11.3.21 RemoveRegistration

```
xCommand RemoveRegistration <regid>
```

Removes the specified registration.

### 11.3.22 SubZoneAdd

```
xCommand SubZoneAdd <name> <address> <prefixlength> <totalmode> <total> <percallmode>
<percall>
```

Adds and configures a new subzone.

| | |
|---|---|
| *name* | User assigned label for the subzone. |
| *address* | IP address for the sub-zone. |
| *prefix* | Number of bits which must match for an IP address to be in this subzone. |
| *totalmode* | Determines whether bandwidth is controlled for this node. `None` prevents any calls, `Limited` imposes bandwidth limits, `Unlimited` imposes no bandwidth limits |

### 11.3.23 SubZoneDelete

```
xCommand SubZoneDelete <index>
```

Deletes the indexed subzone.

### 11.3.24 TraversalZoneAdd

```
xCommand TraversalZoneAdd
```

Creates a new traversal zone, allowing a TANDBERG Gatekeeper to connect to the Border Controller. Up to 50 such zones may be created.

The new zone is pre-configured with a link to the traversal subzone and with a pattern match mode of `AlwaysMatch`.

### 11.3.25 TraversalZoneDelete

```
xCommand TraversalZoneDelete <index>
```

Removes the traversal zone with the specified index.

### 11.3.26 ZoneAdd

```
xCommand ZoneAdd <name> <address>
```

Adds a new zone with the specified name and IP address. E.g. `xCommand ZoneAdd B 10.0.0.30`

The zone is pre-configured with a link to the traversal subzone and a pattern match mode of `AlwaysMatch`.

### 11.3.27 ZoneDelete

```
xCommand ZoneDelete <index>
```

Removes the zone with the specified index.

## 11.4 History

The history root command, `xhistory`, is used to display historical data on the Border Controller.

To list all xhistory commands type:

```
xhistory ?
```

To list all history data, type:

```
xhistory
```

To show a specific set of history data, type:

```
xhistory <name>
```

```
xhistory calls
xhistory calls call <n>
```

Displays history data for up to the last 255 calls handled by the Border Controller. Call entries are added to the Call History on call completion. Call histories are listed in reverse chronological order of completion time.

```
xhistory registrations
xhistory registrations registration <n>
```

Displays history data for up to the last 255 registrations handled by the Border Controller. Registration entries are added to the Registration History on unregistration of H.323 entities. Registration histories are listed in reverse chronological order of unregistration time.

## 11.5 Feedback

The feedback root command, `xfeedback`, is used to control notifications of Events and Status changes on the Border Controller.

A Feedback Expression describes an interesting event or change in status. When a Feedback Expression is registered, a notification will be displayed in the shell for each occurrence of the event described by that Expression. Notifications will continue to be displayed for a given event until the Expression is deregistered.

To list all xfeedback commands type:

```
xfeedback ?
```

To list all currently active feedback expressions, type:

```
xfeedback list
```

To register a feedback expression, type:

```
xfeedback register <expression>
```

To deregister the feedback expression with index ¡n¿, type:

```
xfeedback deregister <n>
```

To deregister all feedback expressions, type: `xfeedback deregister 0`

```
xFeedback Register Status/<Calls/Registrations>
```

Registers for feedback on changes in the chosen Status, e.g.: `xFeedback Register Status/Calls`

To register for all Status changes, use: `xFeedback Register Status`

```
xFeedback Register History/<Calls/Registrations>
```

Registers for feedback on History, e.g.: `xFeedback Register History/Calls`

To register for all History, use: `xFeedback Register History`

`xFeedback Register Event/<CallAttempt/Connected/Disconnected/ConnectionFailure/Registration/Unregistration/Bandwidth/ResourceUsage>`

Registers for feedback on the occurrence of the chosen Event, e.g.: `xFeedback Register Event/CallAttempt`

To register for all available Events, use: `xFeedback Register Event`

Registering for the `ResourceUsage` event will return the entire ResourceUsage structure every time one of the ResourceUsage fields changes. ResourceUsage fields consist of:

- Registrations
- MaxRegistrations
- PortRegistrations
- MaxPortRegistrations
- TraversalCalls
- MaxTraversalCalls
- TotalTraversalCalls

## 11.6 Other commands

### 11.6.1 About

`about`

About provides information about the software version installed on the system.

### 11.6.2 Clear

`clear [eventlog/history]`

Clears the event log or history of all calls and registrations.

### 11.6.3 Eventlog

`eventlog eventlog [n/all]`

Displays the eventlog containing information about past events which may be useful for diagnostic purposes.

| | |
|---|---|
| n | The number of lines from end of event log to dump. |
| all | Dumps the whole event log. |

### 11.6.4 Relkey

`relkey`

Displays the release key that this software has been installed with.

### 11.6.5 Syslog

`syslog <level> [ipaddr] [ipaddr]`

Enables tracing to the console.

| | |
|---|---|
| level | Specifies the detail at which to trace. 0-3, 3 gives most logging. |
| ipaddr | Specify up to 10 IP addresses to log information for, all if none specified. |

Setting `syslog 0` will turn off tracing.

# A Appendix: Configuring DNS Servers

In the examples below, we set up an SRV record to handle H.323 URIs of the form *user@example.com* These are handled by the system with the fully qualified domain name of *gatekeeper1.example.com* which is listening on port 1719, the default registration port.

It is assumed that an A record already exists for *gatekeeper1.example.com*. If not, you will need to add one.

## A.1 Microsoft DNS Server

It is possible to add the SRV record using either the command line or the MMC snap in. To use the command line: on the DNS server open a command window and enter

dnscmd . /RecordAdd domain service_name SRV service_data

Where domain is the domain into which you wish to insert the record, service_name the name of the service you're adding and service_data the priority, weight, port and server providing the service as defined by RFC 2782. For example:

dnscmd . /RecordAdd example.com _h323ls._udp SRV 1 0 1719 gatekeeper1.example.com

### A.1.1 BIND 8 & 9

BIND is a commonly used DNS server on UNIX and Linux systems. Configuration is based around two sets of text files: named.conf which describes which zones are represented by the server and a selection of zone files which describe the detail of each zone.

BIND is sometimes run chrooted for increased security. This gives the program a new root directory, which means that the configuration files may not appear where you expect them to be. To see if this is the case on your system, run

```
ps aux grep named
```

This will give the command line that named (the BIND server) was invoked with. If there is a -t option, then the path following that is the new root directory and your files will be located relative to that root.

In /etc/named.conf look for a directory entry within the options section. This will give the directory in which the zone files are stored, possibly relative to a new root directory. In the appropriate zone section, a file entry will give the name of the file containing the zone details.

For more details of how to configure BIND servers. and the DNS system in general see [6]

## A.2 Verifying the SRV record

There are a range of tools available to investigate DNS records. One commonly found on Microsoft Windows and UNIX platforms is *nslookup*. Use this to verify that everything is working

as expected.

```
nslookup -querytype=srv _h323ls._udp.example.com
```

and check the output.

# B Appendix: Configuring LDAP Servers

## B.1 Microsoft Active Directory

### B.1.1 Prerequisites

These comprehensive step by step instructions assume that Active Directory is installed. For details on installing Active Directory please consult your Windows documentation. The following instructions are for Windows Server 2003 Enterprise Edition, if you are not using this version of Windows, your instructions may vary.

The following ITU specifications describe the schemas which are required to be installed on the Active Directory server:

**H.350** Directory services architecture for multimedia conferencing - An LDAP schema to represent endpoints on the network.

**H.350.1** Directory services architecture for H.323 - An LDAP schema to represent H.323 endpoints.

**H.350.2** Directory services architecture for H.235 - An LDAP schema to represent H.235 elements.

The schemas can be downloaded in ldif format from the web interface on the Border Controller. To do this, navigate to the *Border Controller Configuration* → *Files* page and click on the links for the schemas. Copy the downloaded schemas to the Active Directory server.

Open a command prompt and for each file execute the following command:

```
ldifde -i -c DC=X <ldap_base> -f filename.ldf
```

Where <ldap_base> is the base DN for your Active Directory server.

### B.1.2 Adding H.350 objects

**Create the organizational hierarchy**

Open up the Active Directory Users and Computers MMC snap-in. Under your base DN right click and select New → Organizational Unit. Create an Organizational unit called h350.

**NOTE** It is good practice to keep the H.350 directory in its own organizational unit to separate out H.350 objects from other types of objects. This allows access controls to be setup which only allow the Border Controller read access to the BaseDN and therefore limit access to other sections of the directory.

**Add the H.350 objects**

Create an ldif file with the following contents:

```
# MeetingRoom1 endpoint
dn: commUniqueId=comm1,ou=h350,dc=my-domain,dc=com
objectClass: commObject
```

```
objectClass: h323Identity
objectClass: h235Identity
commUniqueId: comm1
h323Identityh323-ID: MeetingRoom1
h323IdentitydialedDigits: 626262
h235IdentityEndpointID: meetingroom1
h235IdentityPassword: mypassword
```

Add the ldif file to the server using the command:

```
ldifde -i -c DC=X <ldap_base> -f filename.ldf
```

This will add a single H.323 endpoint with an H.323 Id alias of *MeetingRoom1* and an E.164 alias of *626262*. The entry also has H.235 credentials of id *meetingroom1* and password *mypassword* which are used during authentication.

### B.1.3 Securing with TLS

To enable Active Directory to use TLS, you must request and install a certificate on the Active Directory server. The certificate must meet the following requirements:

- Be located in the Local Computer's Personal certificate store. This can be seen using the Certificates MMC snap in.
- Have the private details on how to obtain a key associated for use with it stored locally. When viewing the certificate you should see a message saying “You have a private key that corresponds to this certificate”.
- Have a private key that does not have strong private key protection enabled. This is an attribute that can be added to a key request.
- The Enhanced Key Usage extension includes the Server Authentication object identifier, again this forms part of the key request.
- Issued by a CA that both the domain controller and the client trust.
- Include the Active Directory fully qualified domain name of the domain controller in the common name in the subject field and/or the DNS entry in the subject alternative name extension.

## B.2 OpenLDAP

### B.2.1 Prerequisites

These instructions assume that an OpenLDAP server has already been installed. For details on installing OpenLDAP see the documentation at http://www.openldap.org.

The following examples use a standard OpenLDAP installation on the Linux platform. For installations on other platforms the location of the OpenLDAP configuration files may be different. See the OpenLDAP installation documentation for details.

### B.2.2 Installing the H.350 schemas

The following ITU specification describes the schemas which are required to be installed on the LDAP server:

**H.350** Directory services architecture for multimedia conferencing - An LDAP schema to represent endpoints on the network.

**H.350.1** Directory services architecture for H.323 - An LDAP schema to represent H.323 endpoints.

**H.350.2** Directory services architecture for H.235 - An LDAP schema to represent H.235 elements.

The schemas can be downloaded in ldif format from the web interface on the Border Controller. To do this, navigate to the *Border Controller Configuration* → *Files* page and click on the links for the schemas.

Copy the downloaded schemas to the OpenLDAP schema directory:

```
/etc/openldap/schemas/commobject.ldif
/etc/openldap/schemas/h323identity.ldif
/etc/openldap/schemas/h235identity.ldif
```

Edit /etc/openldap/slapd.conf to add the new schemas. You will need to add the following lines:

```
include /etc/openldap/schemas/commobject.ldif
include /etc/openldap/schemas/h323identity.ldif
include /etc/openldap/schemas/h235identity.ldif
```

The OpenLDAP daemon (slapd) must be restarted for the new schemas to take effect.

### B.2.3 Adding H.350 objects

#### Create the organizational hierarchy

Create an ldif file with the following contents:

```
# This example creates a single organisational unit to contain
# the H.350 objects
dn: ou=h350,dc=my-domain,dc=com
objectClass: organizationalUnit
ou: h350
```

Add the ldif file to the server using the command:

```
slapadd -l <ldif_file>
```

This organizational unit will form the BaseDN to which the Border Controller will issue searches. In this example the BaseDN will be *ou=h350,dc=my-domain,dc=com*.

**NOTE** It is good practice to keep the H.350 directory in its own organizational unit to separate out H.350 objects from other types of objects. This allows access controls to be setup which only allow the Border Controller read access to the BaseDN and therefore limit access to other sections of the directory.

#### Add the H.350 objects

Create an ldif file with the following contents:

```
# MeetingRoom1 endpoint
dn: commUniqueId=comm1,ou=h350,dc=my-domain,dc=com
objectClass: commObject
objectClass: h323Identity
objectClass: h235Identity
commUniqueId: comm1
h323Identityh323-ID: MeetingRoom1
h323IdentitydialedDigits: 626262
h235IdentityEndpointID: meetingroom1
h235IdentityPassword: mypassword
```

Add the ldif file to the server using the command:

```
slapadd -l <ldif_file>
```

This will add a single H.323 endpoint with an H.323 Id alias of *MeetingRoom1* and an E.164 alias of *626262*. The entry also has H.235 credentials of id *meetingroom1* and password *mypassword* which are used during authentication.

### B.2.4 Securing with TLS

The connection to the LDAP server can be encrypted by enabling Transport Level Security (TLS) on the connection. To do this you must create an X.509 certificate for the LDAP server to allow the Border Controller to verify the server's identity. Once the certificate has been created you will need to install the following three files associated with the certificate onto the LDAP server:

- The certificate for the LDAP server.
- The private key for the LDAP server.
- The certificate of the Certificate Authority (CA) that was used to sign the LDAP server's certificate.

All three files should be in PEM file format.

The LDAP server must be configured to use the certificate. To do this, edit /etc/openldap/slapd.conf and add the following three lines:

```
TLSCACertificateFile <path to CA certificate>
TLSCertificateFile <path to LDAP server certificate>
TLSCertificateKeyFile <path to LDAP private key>
```

The OpenLDAP daemon (slapd) must be restarted for the TLS settings to take effect.

For more details on configuring OpenLDAP to use TLS consult the OpenLDAP Administrator's Guide.

To configure the Border Controller to use TLS on the connection to the LDAP server you must upload the CA's certificate as a trusted CA certificate. To do this, navigate to the *Border Controller Configuration* → *Files* page and upload the certificate.

# C Approvals

The product has been approved by various international approval agencies, among others: UL and Nemko. According to their Follow-Up Inspection Scheme, these agencies also perform production inspections at a regular basis, for all production of TANDBERG's equipment.

The test reports and certificates issued for the product show that the TANDBERG Border Controller, Type number TTC2-02, complies with the following standards.

## EMC Emission - Radiated Electromagnetic Interference

- EN55022:1994 + A1:1995 + A2:1997 Class A.
- FCC Rules and Regulations 47CFR, Part 2, Part 15.
- CISPR PUB.22 Class A

## EMC Immunity

- EN 55024:1998 + A1:2001
- EN 61000-3-2:2000
- EN 61000-3-3:1995 + A1:2001

## Electrical Safety

- IEC 60950 3rd edition 1999
- EN 60950 3rd edition 2000
- UL 60950 3. Edition
- CSA C22.2 No. 950-M95

# D Technical Specifications

## System Capacity

500 registered traversal endpoints
100 traversal calls
100 zones

Option keys may restrict the system to a lower capacity than specified above.

## Ethernet Interfaces

3 x LAN/Ethernet (RJ-45) 10/100 Base-TX (2 disabled)

## System console port

2 x COM ports (front and rear), RS-323 DB-9 connector 2 x USB (disabled)

## ITU standards

ITU-T H.323 version 4 including Annex O
ITU-T H.460.18, H.460.19
ITU-T H.235
ITU-T H.350

## Security Features

IP Administration passwords
Management via SSH and HTTPS
Software upgrade via HTTPS and SCP

## System Management

Configuration via serial connection, Telnet, SSH, HTTP, HTTPS
Software upgraded via HTTP, HTTPS and SCP

## Environmental Data

Operation temperature: 0°C to 35°C (32°F to 95°F)
Relative humidity: 10% to 90% non-condensing

## Physical Dimensions

Height: 4.35 cm (1.72 inches)
Width: 42.6 cm (16.8 inches)
Depth: 22.86 cm (9 inches)
1U rack mounted chassis

## Power supply

90   264V full range @47   63 Hz

## Certification

LVD 73/23/EC
EMC 89/366/ECC

# References


[1] ITU Specification: H.235 Security and encryption for H-Series (H.323 and other H.245-based) multimedia terminals

[2] ITU Specification: H.350 Directory services architecture for multimedia conferencing

[3] `http://www.ietf.org/rfc/rfc2782.txt`

[4] `http://www.ietf.org/rfc/rfc3164.txt`

[5] `http://www.ietf.org/rfc/rfc3880.txt`

[6] *DNS and BIND* Fourth Edition Albitz and Liu, O'Reilly and Associates, ISBN: 0-596-00158-4

# E Glossary

**Alias** The name an endpoint registers with the Border Controller. Other endpoints can then use this name to call it.

**ARQ, Admission Request** An endpoint RAS request to make or answer a call.

**E.164** An ITU standard for structured telephone numbers. Each telephone number consists of a country code, area code and subscriber number. For example, TANDBERG's European Headquarters' phone number is +47 67 125125, corresponding to a country code of 47 for Norway, area code of 67 for Lysaker and finally 125125 to determine which phone line in Lysaker.

**LRQ, Location Request** A RAS query between Gatekeepers or Border Controllers to determine the location of an endpoint.

**PortRegistration** A measure of the number of systems and aliases registered with the Border Controller. Each endpoint may register one alias of each type (E.164, H323 etc) and consume 1 PortRegistration in total. If it registers more aliases, e.g. an MCU may register multiple E.164 aliases, then each additional alias counts as an additional PortRegistration. It is the number of PortRegistrations which is controlled by the option key.

**RAS, Registration, Admission and Status Protocol** Protocol used between endpoints and Border Controller to register and place calls.

**Traversal call** An H.323 call which uses a Border Controller. The Border Controller cooperates with the endpoint or TANDBERG gatekeeper to allow communication through a firewall. All signalling and media is routed through the Border Controller.

# Index